

# Assignment # 1

## ATM Software

(Due Date: October 25, 2020 Time: 11:59 PM)

Hello Everyone! The main aims of your first assignment are to:

- (1) Familiarize you with basic C# Syntax.
- (2) To offer as helpful practice for using File System and N-Tier Architecture.
- (3) And yes, to help us judge how much you have learnt in the past weeks.

### Assignment Statement

In this assignment you are going to implement an ATM system from which users can manage their accounts. There are two types of users, Customers and Administrators. Both are presented with their own menus (after login of course). Customers can use the system to withdraw cash, transfer cash from one account to another, deposit cash and get their current balance. Administrators can create, delete, view and update accounts of different users. They must also be able to view certain reports about users and accounts. All data should be stored in a File system.

### Content / Requirement

When your program starts, it should be displaying a login screen. User will be asked to enter a login and 5 digit pin code. The system verifies the login and pin and displays an error if it is incorrect. If the user types the pin code incorrectly three times consecutively then the system should **disable** that login until further notice (i.e. the Administrator changes the status of the user).

*Note: The writing in green is entered by the user and input must be through command line.*

e.g.

Enter login: Adnan123  
Enter Pin code: 12345

## Customer Menu

A customer is then taken to the customer option menu where he will select one of the following options;

1----Withdraw Cash  
2----Cash Transfer  
3----Deposit Cash  
4----Display Balance  
5----Exit

Please select one of the above options:

### 1----Withdraw Cash

The user must be displayed a menu to select the mode of withdrawal as follows:

- a) Fast Cash
- b) Normal Cash

Please select a mode of withdrawal:

### a) Fast Cash

In case of fast cash the user must be presented with a menu such as the one below and asked to choose one of the predefined denominations of money. If he chooses withdraw, the user is asked to select the amount from the options given. No matter which option the user uses to withdraw the money the system must check that the amount is valid (i.e. there is enough money in the account).

1----500  
2----1000  
3----2000  
4----5000  
5----10000  
6----15000  
7----20000

Select one of the denominations of money: 1  
Are you sure you want to withdraw Rs.500 (Y/N)? Y

Cash Successfully Withdrawn!  
Do you wish to print a receipt (Y/N)? Y

Account #12

Date: 14/09/2005

Withdrawn: 500  
Balance: 154500

### **b) Normal Cash**

In case of normal cash, the user should be asked the amount he wishes to withdraw. In this case there is **no** compulsion that the amount should be a multiple of 500. However, the amount must be valid.

Enter the withdrawal amount: 3600

Cash Successfully Withdrawn!  
Do you wish to print a receipt (Y/N)? Y

Account #12  
Date: 14/09/2005

Withdrawn: 3600  
Balance: 154500

To make things a little more interesting, a single account holder cannot withdraw more than 20,000 in one day. (If customer withdraws 15,000 once and tries to withdraw 6,000, customer should not be allowed but customer can withdraw 4,000). Please NOTE that this applies to BOTH Fast Cash and Normal Cash. i.e. the joint withdrawal in both these modes should not exceed 20,000.

### **2----Cash Transfer**

If the user selects transfer cash he is asked to specify the amount in multiples of five hundred that he wishes to transfer. Then he should be asked to enter the account number to which he wishes to transfer the money. The user must be asked to enter the account number twice to make sure he got the correct number. In addition the second time he enters the account number he should also be able to see the name of that account holder to make sure it is the right person.

For example,

Enter amount in multiples of 500: 3500  
Enter the account number to which you want to transfer: 15

You wish to deposit Rs 3,500 in account held by Mr. Usman Ismail; If this information is correct please re-enter the account number: 15

Transaction confirmed.

Do you wish to print a receipt (Y/N)? Y

Account #12  
Date: 14/09/2005

Amount Transferred: 3500  
Balance: 154500

### **3----Deposit Cash**

If the user selects deposit cash he is asked to specify the amount that he wishes to deposit into the account. Of course there is no max limit to how much money is deposited. E.g:

Enter the cash amount to deposit: 12562

Cash Deposited Successfully.  
Do you wish to print a receipt (Y/N)? Y

Account #12  
Date: 14/09/2005

Deposited: 12562  
Balance: 154500

### **4----Display Balance**

This obviously displays the balance on the screen. It is similar to receipt (i.e. it has date account number etc) but it does not have any transactions.  
For example:

Account #12  
Date: 14/09/2005

Balance: 154,500

## **Administrator Menu**

If the user who logs in is an administrator he should be presented with the following menu,

1----Create New Account.  
2----Delete Existing Account.  
3----Update Account Information.  
4----Search for Account.  
5----View Reports  
6----Exit

### **1---Create New Account**

If the user selects this option then he is asked to enter account information. The system should check the validity of the data i.e. type can only be 'Savings' or 'Current'.

```
Login: Javed123
Pin Code: 12345
Holders Name: XYZ
Type (Savings,Current): Savings
Starting Balance: 6000
Status: Active
```

Obviously, an account number has to be assigned to this newly created account. For this purpose, you are required to check the last account number created and add one to it. So, if there exists an account number 18 in the table (and there is no Account # 19) , then 19 should be account number attributed to this account. The administrator must be informed of this after he finishes making the account i.e.

```
Account Successfully Created - the account number assigned
is: 19
```

### **2---Delete Existing Account**

If the user selects this option he is asked to enter an account number. He is then asked confirm that this account should be deleted.

```
Enter the account number to which you want to delete: 15
```

```
You wish to delete the account held by Mr Usman Ismail; If
this information is correct please re-enter the account
number: 15
```

```
Account Deleted Successfully
```

### **3---Update Account Information**

The program should first ask the admin to enter the account number to be updated:

```
Enter the Account Number: 15
```

Then, a menu should be displayed which shows the admin all the old information followed by a step-by-step inquiry to change particular fields. If he leaves a field blank then the old information is propagated i.e. the previous

information remains unchanged.

Account # 15  
Type: Savings  
Holder: Mr James  
Balance: 50,000  
Status: Disabled

Please enter in the fields you wish to update (leave blank otherwise):

Login: dotNet66  
Pin Code: 45678  
Holders Name: *(for example, I leave this blank and simply press enter)*

Status: Active

Your account has been successfully been updated.

#### **4---Search for Account**

This will display a menu asking the admin to enter any search criteria for each field. If the user leaves a field blank then it should not be included in the search i.e. if user does not specify holder name then all accounts matching other criteria with any holder name should be displayed.

SEARCH MENU:

Account ID:  
User ID:  
Holders Name:  
Type (Savings Current): Savings  
Balance: 80,000  
Status: Active

==== SEARCH RESULTS =====

Account ID	User ID	Holders Name	Type	Balance	Status
15	44045	DotNet Dude	Savings	80,000	Active
18	43075	M. XYZ	Savings	80,000	Active

Please note that this search is done by 'AND'ing the search criterias i.e. an account's information should be listed in the search results if an only if all the search criteria's are met.

## 5---View Reports

If the user selects view reports then the he should be asked to specify one of two reports; For the first (by amount) the user will specify a minimum and maximum range for the balance of an account. All accounts with balance in between the range must be displayed in a similar fashion to the one shown above in search for accounts. In the second option the system shows all transactions made by a specified customer within a specified date range (inclusive). i.e. The user enters an account number and a minimum and maximum date. All transactions between the two dates need to be shown.

1---Accounts By Amount  
2---Accounts By Date

For example, if the user enters 1:

Enter the minimum amount: 12500  
Enter the maximum amount: 190112

==== SEARCH RESULTS =====

Account ID	User ID	Holder's Name	Type	Balance	Status
11	44045	Pervaiz M	Savings	12,500	Active
24	43075	Bhutto Benazir	Savings	190,111	Active

Or if the user enters 2:

*Note: The format for the date must always be Day/Month/Year [DD/MM/YYYY]*

Enter the starting date: 12/12/2005  
Enter the ending date: 19/12/2005

==== SEARCH RESULTS =====

Transaction Type	User ID	Holder's Name	Amount	Date
Cash Withdrawal	44045	Bob Jack	2,500	17/12/2005

## **Design Cues and Requirements**

- Use File systems to store all data.

- Make sure you separate the implementation from the interface completely using N-Tier architecture because we are going to ask you to give the same assignment with GUI features later on.
- All data entry points should have proper error checks and error messages.
- All data entry is through command line interface.
- There must be proper commenting throughout your code.
- **We expect that your system should be very robust, so apply intelligent checks; in no case should your program crash or produce undesirable results.**
- Use Function and Variable names intelligently. The harder your code is to check, the higher are the chances of you getting lower marks. If your program crashes for some reason during checking, I will be just scrolling through your code and give you marks on the quality of the code. So, it's purely in your interest to use nice functions and variables names.

### **One more Functionality**

However the login information must be **encrypted** when it is stored to disk and **decrypted** when it is needed. We are going to use a very simple encryption technique which is as follows.

For alphabets we swap A with Z, B with Y and so on.  
 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z  
 Z Y X W V U T S R Q P O N M L K J I H G F E D C B A

For Number we have  
 0123456789  
 9876543210



*Well that's all what you have to do till Deadline.*

### Marking Scheme

Item	Marks
Correct Interface (error checking of extreme values)	10%
Comments	5%
Encryption & Decryption	10%
Log in (with disable account)	5%
Deposit Money	5%
Print Receipt	2%
Withdraw & Transfer cash	10%
Create Update Delete Accounts	5% + 5% + 5%
Searching for Accounts	5%
Account Reports	10 % (5% each)
N Tier Architecture with proper modular approach and there is proper separation of codes into classes and methods. No global variables etc.	23%

GOOD LUCK :)