**PROJECT Bank Account Creation – *More* on Sequential Control 100 points**

**Objective** To type a simple Java program, execute ( run ) the program for some particular values, observe the output and then modify the program.

***PROJECT DESCRIPTION***

Type, compile and run the basic Java program that is shown in **Figure 1** , which follows.

Then compile and run your program, observe the output then modify the program.

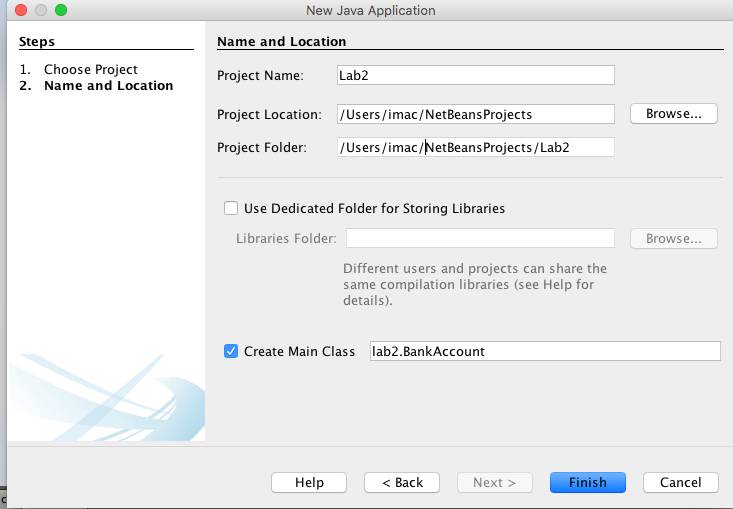
***Information About This Project***

For this project we will track the bank account balance of the user entering in banking data such as for deposits and withdrawals.

***Steps To Complete This Project***

**STEP 1**  **Open NetBeans**

Open NetBeans and create a Java project with the following details.



Of course your Project Location and Folder can be set to either the default locations or to paths of you desired choices.

In your **Code** window, shown below, copy in the program code shown in **Figure 1** below, in the appropriate places, except substitute your own name in place of Sammy Student.

|  |
| --- |
|  |

**Figure 1 Source Code for the BankAccount Program**

|  |
| --- |
| /\*  \* Program for BankAccount data including deposits and withdrawals.  \* Running balance is kept along the way and displayed at program end  \*/  package lab2;  import java.util.Scanner;  /\*\*  \* @author Sammy Student  \*/  public class BankAccount {  /\*\*  \* @param args the command line arguments  \*/  public static void main(String[] args) {  // programs local variables  int acctNum = 0;  double acctBal = 0.0, currency = 0.0;  double checks = 0.0, coins = 0.0;  double newBal = 0.0, withdraw = 0.0;  Scanner sc = new Scanner(System.in);  // the program header  System.out.println("<< Bank Account >>");  System.out.println("------------------");  System.out.println(" ");  // prompt user for data and receive data  System.out.println("please enter your account number");  acctNum = sc.nextInt();  System.out.println("your account number " + acctNum);  System.out.println("please enter your account balance");  acctBal = sc.nextDouble();  System.out.printf("\*\*account balance %6.2f\n" , acctBal);  System.out.println("please enter the cash deposit amount");  currency = sc.nextDouble();  System.out.printf("\*\*deposit amount (cash) $%6.2f\n" ,currency);  System.out.println("please enter the check deposit amount");  checks = sc.nextDouble();  System.out.printf("\*\*deposit amount (checks) $%6.2f\n" ,checks);  System.out.println("please enter the coins deposit amount");  coins = sc.nextDouble();  System.out.printf("\*\*deposit amount (coins) $%6.2f\n" ,coins);  System.out.println("please enter the withdrawal amount");  withdraw = sc.nextDouble();  System.out.printf("\*\*withdrawal amount (cash) $%6.2f\n" ,withdraw);  // the program footer  System.out.println(" ");  System.out.println("thank you!");  System.out.println("------------------");  }  } |

**STEP 2 Build, compile and Run the Program**

From the NetBeans Run menu select Run Project (Lab2) to run your app.

**STEP 3 Test the Program**

Once you have successfully compiled your program, enter the following information, when prompted, into the output **Console** window of NetBeans.

**Initial Test Run**

please enter your account number

1234

your account number 1234

please enter your account balance

100

\*\*account balance 100.00

please enter the cash deposit amount

20

\*\*deposit amount (cash) $ 20.00

please enter the check deposit amount

30

\*\*deposit amount (checks) $ 30.00

please enter the coins deposit amount

2.50

\*\*deposit amount (coins) $ 2.50

please enter the withdrawal amount

100

\*\*withdrawal amount (cash) $100.00

**STEP 4 Verify Your Output**

You’ll notice a major issue hopefully when you have checked over your code logic and syntax when you ran through your program with inputs, namely the last entry where the user was prompted for a withdrawal amount. There really was no check and balances in place in the code as the user could of entered any amount for withdrawal and that would not be logistically correct!

**STEP 5 Modify Your Program**

Modify your existing code to allow the user to see a running balance of all the deposits entered in as well as the initial account balance entered *before* the user is asked for a withdrawal amount. Use only existing variables to obtain and display a running balance.

Also at the end of the program, *after* the withdrawal amount has been entered in, also display the new balance in the users account.

A running sample follows that can be used to test exact input values on your programs rerun. Make sure to verify the ending balances along the way in the sample that follows. Your output should match the same.

**Modified Run**

please enter your account number

1000

your account number 1000

please enter your account balance

100

account balance 100.00

please enter the cash deposit amount

200

deposit amount (cash) $200.00

please enter the check deposit amount

300

deposit amount (checks) $300.00

please enter the coins deposit amount

2.50

deposit amount (coins) $ 2.50

**new account balance $602.50**

please enter the withdrawal amount

150

withdrawal amount (cash) $150.00

**new account balance $452.50**

**STEP 6 Submit Your Project**

Once you have determined that your modified program is correctly computing the bank balances of a given user, complete the submission process as follows:

Open MS Word and type a heading for a new document that includes your full name, course number, lab number and date.

Within the document paste in a snapshot of your modified code. Label your snapshot of your modified run with a reasonable description.

After your snapshot, paste in your finished source code as well copied in from your NetBeans editor.

Submit your MS Word document to Blackboard when complete.