Sidak Manchanda

**Pseudocode for Class UserInterface**

*Extends JFrame and Implements ActionListener*

Create the following as JTextFields: textFieldName, textFieldPhone, textFieldStreet, textFieldPostalCode, textFieldBalance, textFieldWithdraw, textFieldSavingsBalance, textFieldSavingsWithdraw ,textFieldSavingsDeposit , textFieldDeposit

Create the following as JButtons btnAddNewCustomer,btnUpdateExistingCustomer, btnInitiateTransaction, btnSavingsInitiateTransaction and btnClear

Declare the following JToggleButtons tglbtnEnableSavings,tglbtnEnable

Declare a JComboBox province

Declare JPanels chsCus, innerchsCus;

Declare the JLabels lblCityOutput, lblAccountNumberOutput,lblAccountNumber, lblSavingsAccountNumber

JPanel chsCus, innerchsCus;

Create an array list of Customers, JButtons, Savings and Chequing

Declare and initialize int currentCustomer and numCus as 0

**Main method**

Create and initialize new userInterface Object

Show a message dialog telling user to add new customer after filling the personal information

Set the userInterface to be visible

**End Main**

**Class Constructor**

Initialize all objects that were only declared in the beginning of the class

Set the default close operation to exit on close

Set the Frame’s Layout to a Grid Bag Layout

Add in the chscus to the JFrame

Add on innerchsCus to chsCus

Set the layout for innerchsCus to a GridBagLayout

Create and add a new JLabel with text “Customers” to innerchsCus

Create and initialize new JPanel called information

Set information’s layout manager to gridbaglayout

Add information to the frame

Create and initialize new JPanel called customer

Set customer’s layout manager to gridbaglayout

Add customer to information

Create and add new JLabel with text “Customer Information” to customer

Add the labels “Name”, “Phone Number”, “Street”, “Postal Code” , ‘Province” and “City” to customer

Add the textfields textFieldName, textFieldPhone, textFieldStreet, textFieldPostalCode, as well as the province JCombobox and lblCityOutput to customer panel

Add the JButtons btnAddNewCustomer,btnUpdateExistingCustomer and btn clear to the customer panel

Add action listeners to btnAddNewCustomer,btnUpdateExistingCustomer and btn clear to the customer panel and province(JComboBox)

Add all provinces as options on the combobox

Create and initialize a new panel called accounts

Add accounts to the information panel

Create, Initialize and add two JPanels chequingPanel and SavingsPanel to accounts

Set layout of above three panels to GridBagLayout

Add Label “Chequing” to chequing panel

Add tglbtnEnable to Chequing Panel

Add labels “Balance:”, “Acc.Num:” , “Withdraw:” and “Deposit:” to the chequing panel

Add TextFields textFieldBalance, textFieldWithdraw, textFieldDeposit and label lblAccountNumber to the chequing panel

Add Button btnInitiateTransaction to the chequing panel and add an action listener to that button

Add Label “Savings” to savings panel

Add tglbtnEnableSavings to Savings Panel

Add labels “Balance:”, “Acc.Num:” , “Withdraw:” and “Deposit:” to the savings panel

Add TextFields textFieldSavingsBalance, textFieldSavingsWithdraw, textFieldSavingsDeposit and label lblAccountNumberOutput to the chequing panel

Add Button btnInitiateTransactionSavings to the savings panel and add an action listener to that button

Use gridBagconstarints to organize all panels and objects

Set the size of the frame to 1000\*500

**End Constructor**

**Method actionPerformed(Actionevent e) – Actionlistener**

**If** the source of the source of the change is the JcomboBox Province

**If** the selected item is \_\_\_\_\_ province

Then set the lblCity output to the capital city \_\_\_\_

**End if**

**Else if** the source of the click is add a new customer

**If** the customer name field is not blank

Set the currentCustomer equal to numCus

Add a new customer to the customer array and pass on the values obtained from the textfields

Add a new savings and chequing account to the appropriate arraylists with the balance as zero and pass the current customer as the owner

Create a new Jbutton temp

Set its name to the name of the current customer

Add an action listener to temp

Set the gridbagconstraints for temp

Add the button to the arraylist of buttons called savedCustomers

Add the button in savedCustomers to the side panel called innerchsCus

Update the frame

Add one to numCus

Disable all the textfields and buttons in the savings and chequing panel

Deselect both the togglebuttons tglbtnEnable and tglbtnEnableSavings

**End if**

**Else if-** the source is btnUpdateExistingCustomer

Get the currentCustomer’s info from the arraylist Customers

Use the set\_\_\_\_ method to set the \_\_\_\_ retrieved from textField\_\_\_\_

Use the setProvince method to set the province retrieved from the JComboBox province

**Else if-**  if the source is the enable button for the chequing panel

Enable textFieldBalance,textFieldWithdraw, textFieldDeposit,btnInitiateTransaction

Set text in textFieldBalance to “0”

**Else if-**  if the source is the enable button for the savings panel

Enable textFieldSavingsBalance,textFieldSavingsWithdraw,textFieldSavingsDeposit, btnInitiateTransactionSavings

Set text in textFieldSavingsBalance to “0”

**Else if-**  the btninitiatetransaction in chequing panel is clicked

Get the chequing account object from the arraylist custmerChequing at the index of currentCustomer

Get and set the balance from the textFieldBalance

Create a boolean called valid.

Use the current customerChequing object and withdraw the amount in textFieldWithdraw(Valid will be assigned the value returned from withdraw)

**If**  valid is false and the withdrawal amount is greater than zero

Display a message dialog with “Invalid Transaction”

**End if**

Use the current customerChequing object and deposit the amount in textFieldDeposit

Set the new Balance in the textField to the balance retrieved from the object

Set the lblAccountNumber to the account number retrieved from the customerChequing chequing object

Set the textFieldWithdraw and textFieldDeposit to “0”

**Else if-**  the btninitiatetransactionSavings in the savings panel is clicked

Get the savings account object from the arraylist custmerSavings at the index of currentCustomer

Get and set the balance from the textFieldSavingsBalance

Create a boolean called valid.

Use the current customerSavings object and withdraw the amount in textFieldSavingsWithdraw (Valid will be assigned the value returned from withdraw)

**If**  valid is false and the withdrawal amount is greater than zero

Display a message dialog with “Invalid Transaction”

**End if**

Use the current customerSavings object and deposit the amount in textFieldSavingsDeposit

Set the new Balance in the textField to the balance retrieved from the object

Set the lblSavingsAccountNumber to the account number retrieved from the customerSavings Savings object

Set the textFieldSavingsWithdraw and textFieldSavingsDeposit text to “0”

**Else if**- is the clear button is clicked

Set the text for textFieldName, textFieldPhone, textFieldStreet, textFieldPostalCode to “”

Set the province combo box’s selected item to “”

Set the text for lblCityOutput to “”

**Else-** If none of these are the source the source must be the buttons on th side panel

Declare and initialize int I to equal to the index of the button clicked in savedCutomers(the JButton ArrayList)

Set the currentCustomer to equal i

Set the textFieldName to the name retrieved from the Customers arraylist at the index of i

Set the textFieldPhone to the phone retrieved from the Customers arraylist at the index of i

Set the textFieldStreet to the street retrieved from the Customers arraylist at the index of i

Set the textFieldPostalCode to the postal code retrieved from the Customers arraylist at the index of i

Set the lblCityOutput to the city retrieved from the Customers arraylist at the index of i

**If** the province retrieved from the Customers arraylist at the index of I is “\_\_\_\_\_\_”

Then set the selected item in province comboBox to “\_\_\_\_\_\_\_\_”

**End if**

**If** the balance of the customerChequing at the index I is equal to zero

Then disable all the textfields and buttons in the chequing panel

**Else**

Then enable all the textfields and buttons in the chequing panel

Set the balance and account number to that retrieved from the customerChequing at the index of i

Set the text in textFieldWithdraw and textFieldDeposit to 0

**End if**

**If** the balance of the customerSavings at the index I is equal to zero

Then disable all the textfields and buttons in the Savingsg panel

**Else**

Then enable all the textfields and buttons in the Savings panel

Set the balance and account number to that retrieved from the customerSavings at the index of i

Set the text in textFieldSavingsWithdraw and textFieldSavingsDeposit to 0

**End if**

**End if**

**End ActionPerformed**