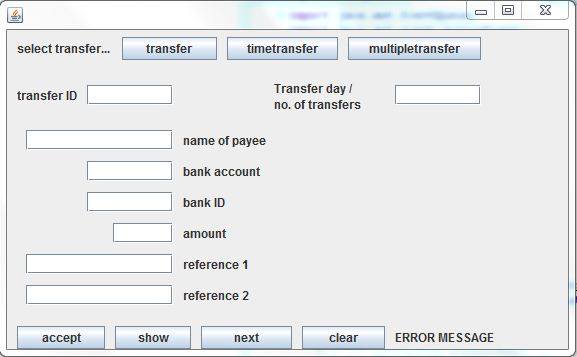


**GUI**



**Source code:**

**User.java**

**public** **class** User {

**private** MultipleTransfer multTransfer;

**private** TimeTransfer timeTransfer;

**private** Transfer transfer;

**public** User() {

**this**.multTransfer = **new** MultipleTransfer();

**this**.timeTransfer = **new** TimeTransfer();

**this**.transfer = **new** Transfer();

}

**public** MultipleTransfer getMultTransfer() {

**return** **this**.multTransfer;

}

/\*\*\* **@param** multTransfer \*/

**public** **void** setMultTransfer(MultipleTransfer multTransfer) {

**this**.multTransfer = multTransfer;

}

**public** TimeTransfer getTimeTransfer() {

**return** **this**.timeTransfer;

}

/\*\* **@param** timeTransfer \*/

**public** **void** setTimeTransfer(TimeTransfer timeTransfer) {

**this**.timeTransfer = timeTransfer;

}

**public** Transfer getTransfer() {

**return** **this**.transfer;

}

/\*\* **@param** transfer \*/

**public** **void** setTransfer(Transfer transfer) {

**this**.transfer = transfer;

}

}

**Transfer.java**

**public** **class** Transfer {

**protected** **long** transfer\_id;

**protected** String nameOfPayee;

**protected** **long** accNumber;

**protected** **long** bankid;

**protected** **float** amount;

**protected** String reference1;

**protected** String reference2;

**public** Transfer(){

}

**public** Transfer(**long** transfer\_id, String nameOfPayee, **long** accNumber, **long** bankid, **float** amount, String ref1, String ref2){

**this**.transfer\_id = transfer\_id;

**this**.nameOfPayee = nameOfPayee;

**this**.accNumber = accNumber;

**this**.bankid = bankid;

**this**.amount = amount;

**this**.reference1 = ref1;

**this**.reference2 = ref2;

}

**public** **long** getTransfer\_id() {

**return** transfer\_id;

}

**public** **void** setTransfer\_id(**long** transfer\_id) {

**this**.transfer\_id = transfer\_id;

}

**public** String getNameOfPayee() {

**return** nameOfPayee;

}

**public** **void** setNameOfPayee(String nameOfPayee) {

**this**.nameOfPayee = nameOfPayee;

}

**public** **long** getAccNumber() {

**return** accNumber;

}

**public** **void** setAccNumber(**long** accNumber) {

**this**.accNumber = accNumber;

}

**public** **long** getBankid() {

**return** bankid;

}

**public** **void** setBankid(**long** bankid) {

**this**.bankid = bankid;

}

**public** **float** getAmount() {

**return** amount;

}

**public** **void** setAmount(**float** amount) {

**this**.amount = amount;

}

**public** String getReference1() {

**return** reference1;

}

**public** **void** setReference1(String reference1) {

**this**.reference1 = reference1;

}

**public** String getReference2() {

**return** reference2;

}

**public** **void** setReference2(String reference2) {

**this**.reference2 = reference2;

}

}

**TimeTransfer.java**

**import** java.util.Date;

**public** **class** TimeTransfer **extends** Transfer {

**private** Date date;

**public** TimeTransfer() {

**super**();

}

**public** TimeTransfer(**long** transfer\_id, String nameOfPayee, **long** accNumber, **long** bankid, **float** amount, String ref1, String ref2){

**super**(transfer\_id, nameOfPayee, accNumber, bankid, amount, ref1, ref2);

}

**public** Date getDate() {

**return** date;

}

**public** **void** setDate(Date date) {

**this**.date = date;

}

}

**MultipleTransfer.java**

**public** **class** MultipleTransfer **extends** Transfer {

**private** **int** numberOfTransfers;

**private** Transfer[] transfers;

**public** MultipleTransfer(**int** N) {

**super**();

numberOfTransfers = N;

transfers = **new** Transfer[numberOfTransfers];

}

**public** MultipleTransfer(){

**super**();

numberOfTransfers = 0;

transfers = **new** Transfer[numberOfTransfers];

}

**public** **int** getNumberOfTransfers() {

**return** **this**.numberOfTransfers;

}

/\*\* **@param** numberOfTransfers \*/

**public** **void** setNumberOfTransfers(**int** numberOfTransfers) {

**this**.numberOfTransfers = numberOfTransfers;

}

**public** **void** addTransfer(Transfer transfer){

Transfer[] temp = **new** Transfer[**this**.numberOfTransfers];

temp = transfers;

transfers = **new** Transfer[++**this**.numberOfTransfers];

**for**(**int** i = 0; i< temp.length; i++){

transfers[i] = temp[i];

}

transfers[**this**.numberOfTransfers-1] = transfer;

}

/\*public void setLastTransfer(Transfer temp){

transfers[this.numberOfTransfers-1] = temp;

}\*/

**public** Transfer[] getTransfers() {

**return** transfers;

}

**public** **void** setTransfers(Transfer[] transfers) {

**this**.transfers = transfers;

}

}

**TestFrame.java**

**import** java.awt.EventQueue;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** java.text.ParseException;

**import** java.text.SimpleDateFormat;

**import** java.util.Date;

**import** javax.swing.JButton;

**import** javax.swing.JFrame;

**import** javax.swing.JLabel;

**import** javax.swing.JPanel;

**import** javax.swing.JTextField;

**import** javax.swing.border.EmptyBorder;

**public** **class** TestFrame **extends** JFrame {

**private** JPanel contentPane;

**private** JTextField txtTransferIdInput;

**private** JTextField txtTransferDayInput;

**private** JTextField txtNameInput;

**private** JTextField txtAccInput;

**private** JTextField txtIdInput;

**private** JTextField txtAmountInput;

**private** JTextField txtRefInput;

**private** JTextField txtRefInput\_1;

**private** JLabel lblTransferId;

**private** JLabel lblTransferDay;

**private** JLabel lblTransferIdOutput;

**private** JLabel lblNameOfPayee;

**private** JLabel lblBankAccount;

**private** JLabel lblBankId;

**private** JLabel lblAmount;

**private** JLabel lblReference;

**private** JLabel lblReference\_1;

**private** JLabel lblNameOutput;

**private** JLabel lblAccOutput;

**private** JLabel lblIdOutput;

**private** JLabel lblAmountOutput;

**private** JLabel lblRefOutput;

**private** JLabel lblRefOutput\_1;

**private** JButton btnAccept;

**private** JButton btnShow;

**private** JButton btnNext;

**private** JButton btnClear;

**private** JButton btnMultipletransfer;

**private** JButton btnTimetransfer;

**private** JButton btnTransfer;

**private** JLabel lblSelectTransferType;

**private** JLabel lblErrorMessage;

**private** **boolean** transferPressed;

**private** **boolean** timeTransferPressed;

**private** **boolean** multipleTransferPressed;

**private** **static** User *user*;

/\*\* Launch the application. \*/

**public** **static** **void** main(String[] args) {

EventQueue.*invokeLater*(**new** Runnable() {

**public** **void** run() {

**try** {

TestFrame frame = **new** TestFrame();

frame.setVisible(**true**);

} **catch** (Exception e) {

e.printStackTrace();

}

}

});

*user* = **new** User();

}

/\*\* Create the frame. \*/

**public** TestFrame() {

setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);

setBounds(100, 100, 577, 357);

contentPane = **new** JPanel();

contentPane.setBorder(**new** EmptyBorder(5, 5, 5, 5));

setContentPane(contentPane);

contentPane.setLayout(**null**);

lblSelectTransferType = **new** JLabel("select transfer type");

lblSelectTransferType.setBounds(10, 11, 95, 14);

contentPane.add(lblSelectTransferType);

btnTransfer = **new** JButton("transfer");

btnTransfer.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

transferPressed = **true**;

timeTransferPressed = **false**;

multipleTransferPressed = **false**;

lblTransferDay.setVisible(**false**);

txtTransferDayInput.setVisible(**false**);

btnNext.setVisible(**false**);

}

});

btnTransfer.setBounds(115, 7, 95, 23);

contentPane.add(btnTransfer);

btnTimetransfer = **new** JButton("timetransfer");

btnTimetransfer.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

transferPressed = **false**;

timeTransferPressed = **true**;

multipleTransferPressed = **false**;

lblTransferDay.setVisible(**true**);

txtTransferDayInput.setVisible(**true**);

btnNext.setVisible(**false**);

}

});

btnTimetransfer.setBounds(220, 7, 111, 23);

contentPane.add(btnTimetransfer);

btnMultipletransfer = **new** JButton("multipletransfer");

btnMultipletransfer.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

transferPressed = **false**;

timeTransferPressed = **false**;

multipleTransferPressed = **true**;

lblTransferDay.setVisible(**true**);

txtTransferDayInput.setVisible(**true**);

btnNext.setVisible(**true**);

}

});

btnMultipletransfer.setBounds(341, 7, 133, 23);

contentPane.add(btnMultipletransfer);

lblTransferId = **new** JLabel("transfer ID");

lblTransferId.setBounds(10, 58, 60, 14);

contentPane.add(lblTransferId);

txtTransferIdInput = **new** JTextField();

//txtTransferIdInput.setText("transfer ID input");

txtTransferIdInput.setBounds(80, 55, 86, 20);

contentPane.add(txtTransferIdInput);

txtTransferIdInput.setColumns(10);

lblTransferIdOutput = **new** JLabel("");

lblTransferIdOutput.setBounds(176, 58, 70, 14);

contentPane.add(lblTransferIdOutput);

lblTransferDay = **new** JLabel("<html>Transfer day /<br>no. of transfers");

lblTransferDay.setBounds(267, 41, 111, 51);

contentPane.add(lblTransferDay);

txtTransferDayInput = **new** JTextField();

//txtTransferDayInput.setText("transfer day input");

txtTransferDayInput.setBounds(388, 55, 86, 20);

contentPane.add(txtTransferDayInput);

txtTransferDayInput.setColumns(10);

txtNameInput = **new** JTextField();

//txtNameInput.setText("name input");

txtNameInput.setBounds(19, 100, 147, 20);

contentPane.add(txtNameInput);

txtNameInput.setColumns(10);

txtAccInput = **new** JTextField();

//txtAccInput.setText("acc input");

txtAccInput.setBounds(80, 131, 86, 20);

contentPane.add(txtAccInput);

txtAccInput.setColumns(10);

txtIdInput = **new** JTextField();

//txtIdInput.setText("id input");

txtIdInput.setBounds(80, 162, 86, 20);

contentPane.add(txtIdInput);

txtIdInput.setColumns(10);

txtAmountInput = **new** JTextField();

//txtAmountInput.setText("amount input");

txtAmountInput.setBounds(106, 193, 60, 20);

contentPane.add(txtAmountInput);

txtAmountInput.setColumns(10);

txtRefInput = **new** JTextField();

//txtRefInput.setText("ref1 input");

txtRefInput.setBounds(19, 224, 147, 20);

contentPane.add(txtRefInput);

txtRefInput.setColumns(10);

txtRefInput\_1 = **new** JTextField();

//txtRefInput\_1.setText("");

txtRefInput\_1.setBounds(19, 255, 147, 20);

contentPane.add(txtRefInput\_1);

txtRefInput\_1.setColumns(10);

lblNameOfPayee = **new** JLabel("name of payee");

lblNameOfPayee.setBounds(176, 103, 117, 14);

contentPane.add(lblNameOfPayee);

lblBankAccount = **new** JLabel("bank account");

lblBankAccount.setBounds(176, 134, 117, 14);

contentPane.add(lblBankAccount);

lblBankId = **new** JLabel("bank ID");

lblBankId.setBounds(176, 165, 117, 14);

contentPane.add(lblBankId);

lblAmount = **new** JLabel("amount");

lblAmount.setBounds(176, 196, 117, 14);

contentPane.add(lblAmount);

lblReference = **new** JLabel("reference 1");

lblReference.setBounds(176, 227, 117, 14);

contentPane.add(lblReference);

lblReference\_1 = **new** JLabel("reference 2");

lblReference\_1.setBounds(176, 258, 109, 14);

contentPane.add(lblReference\_1);

lblNameOutput = **new** JLabel("");

lblNameOutput.setBounds(303, 103, 171, 14);

contentPane.add(lblNameOutput);

lblAccOutput = **new** JLabel("");

lblAccOutput.setBounds(303, 134, 156, 14);

contentPane.add(lblAccOutput);

lblIdOutput = **new** JLabel("");

lblIdOutput.setBounds(303, 165, 46, 14);

contentPane.add(lblIdOutput);

lblAmountOutput = **new** JLabel("");

lblAmountOutput.setBounds(303, 196, 138, 14);

contentPane.add(lblAmountOutput);

lblRefOutput = **new** JLabel("");

lblRefOutput.setBounds(303, 227, 161, 14);

contentPane.add(lblRefOutput);

lblRefOutput\_1 = **new** JLabel("");

lblRefOutput\_1.setBounds(303, 258, 161, 14);

contentPane.add(lblRefOutput\_1);

btnAccept = **new** JButton("accept");

btnAccept.addActionListener(**new** ActionListener(){

@Override

**public** **void** actionPerformed(ActionEvent e) {

**if**(txtAccInput.getText().equals("") || txtAmountInput.getText().equals("") || txtIdInput.getText().equals("") || txtNameInput.getText().equals("") || txtTransferIdInput.getText().equals("")){

lblErrorMessage.setText("Please, fill all necessary fields");

**return**;

}

**if**(transferPressed){

**try**{

*user*.setTransfer(**new** Transfer(Long.*valueOf*(txtTransferIdInput.getText()), txtNameInput.getText(), Long.*valueOf*(txtAccInput.getText()), Long.*valueOf*(txtIdInput.getText()), Float.*valueOf*(txtAmountInput.getText()), txtRefInput.getText(), txtRefInput\_1.getText()));

clear();

}

**catch**(NumberFormatException numbException){

lblErrorMessage.setText("Please, fill fields correctly");

}

}

**else** **if**(multipleTransferPressed){

**try**{

*user*.getMultTransfer().addTransfer(**new** Transfer(Long.*valueOf*(txtTransferIdInput.getText()), txtNameInput.getText(), Long.*valueOf*(txtAccInput.getText()), Long.*valueOf*(txtIdInput.getText()), Float.*valueOf*(txtAmountInput.getText()), txtRefInput.getText(), txtRefInput\_1.getText()));

clear();

}

**catch**(NumberFormatException numbException){

lblErrorMessage.setText("Please, fill fields correctly");

}

}

**else** **if**(timeTransferPressed){

**if**(txtTransferDayInput.getText().equals("")){

lblErrorMessage.setText("Set the Date please");

**return**;

}

SimpleDateFormat dateString = **new** SimpleDateFormat("yyyy-MM-dd");

**try**{

Date date = dateString.parse(txtTransferDayInput.getText());

*user*.getTimeTransfer().setDate(date);

}

**catch**(ParseException parseException){

lblErrorMessage.setText("Please set Time as: yyyy-mm-dd");

**return**;

}

clear();

}

}

});

btnAccept.setBounds(10, 296, 88, 23);

contentPane.add(btnAccept);

btnShow = **new** JButton("show");

btnShow.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

showFrame();

}

});

btnShow.setBounds(108, 296, 76, 23);

contentPane.add(btnShow);

btnNext = **new** JButton("next");

btnNext.addActionListener(**new** ActionListener(){

@Override

**public** **void** actionPerformed(ActionEvent e) {

**try**{

Transfer transfer = **new** Transfer(Long.*valueOf*(txtTransferIdInput.getText()), txtNameInput.getText(), Long.*valueOf*(txtAccInput.getText()), Long.*valueOf*(txtIdInput.getText()), Float.*valueOf*(txtAmountInput.getText()), txtRefInput.getText(), txtRefInput\_1.getText());

*user*.getMultTransfer().addTransfer(transfer);

txtTransferDayInput.setText(String.*valueOf*(*user*.getMultTransfer().getNumberOfTransfers()));

clear();

}

**catch**(NumberFormatException numbException){

lblErrorMessage.setText("Please, fill fields correctly");

}

}

});

btnNext.setBounds(194, 296, 91, 23);

contentPane.add(btnNext);

btnClear = **new** JButton("clear");

btnClear.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent e) {

clear();

}

});

btnClear.setBounds(295, 296, 83, 23);

contentPane.add(btnClear);

lblErrorMessage = **new** JLabel("ERROR MESSAGE");

lblErrorMessage.setBounds(388, 300, 163, 14);

contentPane.add(lblErrorMessage);

}

**private** **void** clear(){

lblTransferIdOutput.setText("");

lblNameOutput.setText("");

lblAccOutput.setText("");

lblIdOutput.setText("");

lblAmountOutput.setText("");

lblRefOutput.setText("");

lblRefOutput\_1.setText("");

txtTransferIdInput.setText("");

txtNameInput.setText("");

txtAccInput.setText("");

txtIdInput.setText("");

txtAmountInput.setText("");

txtRefInput.setText("");

txtRefInput\_1.setText("");

txtTransferDayInput.setText("");

}

**private** **void** showFrame(){

lblTransferIdOutput.setText(txtTransferIdInput.getText());

lblNameOutput.setText(txtNameInput.getText());

lblAccOutput.setText(txtAccInput.getText());

lblIdOutput.setText(txtIdInput.getText());

lblAmountOutput.setText(txtAmountInput.getText());

lblRefOutput.setText(txtRefInput.getText());

lblRefOutput\_1.setText(txtRefInput\_1.getText());

}

}