|  |  |
| --- | --- |
| **Logo_Baru-Rasmi_UUM_02** | STIA1123: Programming 2 Semester 191School of Computing, CAS, UUM |
| **PROJECT** | |
| 2/12/2019  Delivery  16/12/2019 | |

This project should contain one **simple login screen** and **main interface** that allow user to manage registration of information. Main interface should allow user to add, delete, update and search data from a file. Data should be loaded from a file using file input output operation. Load the data into a **jtable. Input type should contain at least 5 or more inputs.**

Deliverables:

Save the following into word document

* + UML diagram for class/classes identified
  + Java Classes
  + Print screen of input – process - output

Make sure the source code is readable and the interface screenshot is clear.

Save your word document and upload to:

* UUM Learning
* http://slumberjer.com/teaching/a192/stia1123/upload/

Evaluation will be based on the following:

* **Interface Design**
* **Class Diagram**

|  |  |  |  |
| --- | --- | --- | --- |
| **16** | **263095** | **Muhamad Irwan Ihfan Bin Khalid** | **Order delivery system – Application should be able to store delivery information such as delivery id, type, time, price, address and etc.** |

**UML CLASS DIAGRAM**

|  |
| --- |
| **Order** |
| **- iD :String**  **- type :String**  **- date :String**  **- time :String**  **-price :String**  **-address :String** |
| **+ getID() :String**  **+ getType() :String**  **+ getDate() :String**  **+ getTime() :String**  **+ getPrice() :String**  **+ getAddress() :String**  **+ setDate(String date) : void**  **+ setTime(String time) : void** |

**JAVA MAIN CLASS CODE**

**package ordersystem;**

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

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

**import java.awt.event.MouseAdapter;**

**import java.awt.event.MouseEvent;**

**import java.io.File;**

**import java.awt.FlowLayout;**

**import javax.swing.JTextArea;**

**import java.io.FileReader;**

**import java.io.FileWriter;**

**import java.io.IOException;**

**import java.util.ArrayList;**

**import javax.swing.JButton;**

**import javax.swing.JFrame;**

**import javax.swing.JLabel;**

**import javax.swing.JOptionPane;**

**import javax.swing.JScrollPane;**

**import javax.swing.JTable;**

**import javax.swing.JTextField;**

**import javax.swing.table.DefaultTableModel;**

**public class OrderSystem {**

**ArrayList<Order> orderlist = new ArrayList<Order>();**

**String header[] = new String[]{"ID", "type", "date", "time", "price", "address"};**

**DefaultTableModel dtm = new DefaultTableModel(header, 1);**

**OrderSystem() {**

**JFrame frame = new JFrame("ORDER DELIVERY SYSTEM");**

**frame.setSize(702, 500);**

**JLabel jlabel1 = new JLabel("ID Number :");**

**jlabel1.setBounds(20, 10, 70, 20);**

**frame.add(jlabel1);**

**JLabel jlabel2 = new JLabel("Type :");**

**jlabel2.setBounds(52, 32, 70, 20);**

**frame.add(jlabel2);**

**JLabel jlabel3 = new JLabel("Date :");**

**jlabel3.setBounds(53, 52, 70, 20);**

**frame.add(jlabel3);**

**JLabel jlabel4 = new JLabel("Time :");**

**jlabel4.setBounds(51, 74, 70, 20);**

**frame.add(jlabel4);**

**JLabel jlabel5 = new JLabel("Price :");**

**jlabel5.setBounds(46, 96, 70, 20);**

**frame.add(jlabel5);**

**JLabel jlabel6 = new JLabel("Address :");**

**jlabel6.setBounds(320, 10, 70, 20);**

**frame.add(jlabel6);**

**JTextField jtfiD = new JTextField();**

**jtfiD.setBounds(90, 10, 200, 20);**

**frame.add(jtfiD);**

**JTextField jtfType = new JTextField();**

**jtfType.setBounds(90, 32, 200, 20);**

**frame.add(jtfType);**

**JTextField jtfDate = new JTextField();**

**jtfDate.setBounds(90, 54, 200, 20);**

**frame.add(jtfDate);**

**JTextField jtfTime = new JTextField();**

**jtfTime.setBounds(90, 76, 200, 20);**

**frame.add(jtfTime);**

**JTextField jtfprice = new JTextField();**

**jtfprice.setBounds(90, 98, 200, 20);**

**frame.add(jtfprice);**

**JTextField jtfaddress = new JTextField();**

**jtfaddress.setBounds(380, 10, 200, 20);**

**frame.add(jtfaddress);**

**JButton jbuttonAdd = new JButton("INSERT");**

**jbuttonAdd.setBounds(20, 130, 90, 20);**

**frame.add(jbuttonAdd);**

**JButton jbuttondelete = new JButton("DELETE");**

**jbuttondelete.setBounds(130, 130, 90, 20);**

**frame.add(jbuttondelete);**

**JButton jbuttonsearch = new JButton("SEARCH");**

**jbuttonsearch.setBounds(240, 130, 90, 20);**

**frame.add(jbuttonsearch);**

**JButton jbuttonupdate = new JButton("UPDATE");**

**jbuttonupdate.setBounds(350, 130, 90, 20);**

**frame.add(jbuttonupdate);**

**JButton jbuttonexit = new JButton("EXIT");**

**jbuttonexit.setBounds(200, 330, 90, 20);**

**frame.add(jbuttonexit);**

**JTable jtable = new JTable();**

**jtable.setBounds(10, 166, 800, 150);**

**frame.add(jtable);**

**jtable.setModel(dtm);**

**JScrollPane scrollPane = new JScrollPane(jtable);**

**scrollPane.setBounds(10, 166, 800, 150);**

**new JScrollPane(scrollPane, JScrollPane.VERTICAL\_SCROLLBAR\_AS\_NEEDED, JScrollPane.HORIZONTAL\_SCROLLBAR\_AS\_NEEDED);**

**jtable.setAutoResizeMode(JTable.AUTO\_RESIZE\_OFF);**

**frame.add(scrollPane);**

**jtable.getColumnModel().getColumn(0).setPreferredWidth(70);**

**jtable.getColumnModel().getColumn(1).setPreferredWidth(115);**

**jtable.getColumnModel().getColumn(2).setPreferredWidth(125);**

**jtable.getColumnModel().getColumn(3).setPreferredWidth(100);**

**jtable.getColumnModel().getColumn(4).setPreferredWidth(120);**

**jtable.getColumnModel().getColumn(5).setPreferredWidth(145);**

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

**@Override**

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

**String iD = jtfiD.getText();**

**String type = jtfType.getText();**

**String date = jtfDate.getText();**

**String time = jtfTime.getText();**

**String price = jtfprice.getText();**

**String address = jtfaddress.getText();**

**if (price.length() < 3) {**

**JOptionPane.showMessageDialog(frame, "Price should contain more than 3 char!");**

**return;**

**}**

**if (!isInteger(iD)) {**

**JOptionPane.showMessageDialog(frame, "ID should only contain integer!!!");**

**return;**

**}**

**Order orderdata = new Order(iD, type, date, time, price, address);**

**orderlist.add(orderdata);**

**writeData();**

**}**

**});**

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

**@Override**

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

**String iD = JOptionPane.showInputDialog("ID report to delete?");**

**if (iD != null) {**

**System.out.println("not null");**

**for (int i = 0; i < orderlist.size(); i++) {**

**if (orderlist.get(i).getID().equalsIgnoreCase(iD)) {**

**orderlist.remove(i);**

**}**

**}**

**writeData();**

**}**

**}**

**});**

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

**@Override**

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

**String iD = JOptionPane.showInputDialog("Enter ID Number?");**

**if (iD != null) {**

**for (int i = 0; i < orderlist.size(); i++) {**

**if (orderlist.get(i).getID().equalsIgnoreCase(iD)) {**

**JOptionPane.showMessageDialog(frame, "Found!!!");**

**jtfiD.setText(orderlist.get(i).getID());**

**jtfType.setText(orderlist.get(i).getType());**

**jtfDate.setText(orderlist.get(i).getDate());**

**jtfTime.setText(orderlist.get(i).getTime());**

**jtfprice.setText(orderlist.get(i).getPrice());**

**jtfaddress.setText(orderlist.get(i).getAddress());**

**return;**

**}**

**}**

**JOptionPane.showMessageDialog(frame, "Not Found!!!");**

**}**

**}**

**});**

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

**@Override**

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

**String iD = jtfiD.getText();**

**String type = jtfType.getText();**

**String date = jtfDate.getText();**

**String time = jtfTime.getText();**

**String price = jtfprice.getText();**

**String address = jtfaddress.getText();**

**if (iD != null) {**

**for (int i = 0; i < orderlist.size(); i++) {**

**if (orderlist.get(i).getID().equalsIgnoreCase(iD)) {**

**orderlist.get(i).setDate(date);**

**orderlist.get(i).setTime(time);**

**JOptionPane.showMessageDialog(frame, "Updated!!!");**

**}**

**}**

**}**

**writeData();**

**}**

**});**

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

**@Override**

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

**frame.setVisible(false);**

**frame.dispose();**

**}**

**});**

**readData();**

**jtable.addMouseListener(new MouseAdapter() {**

**@Override**

**public void mouseClicked(MouseEvent e) {**

**int row = jtable.getSelectedRow();**

**jtfiD.setText(dtm.getValueAt(row, 0).toString());**

**jtfType.setText(dtm.getValueAt(row, 1).toString());**

**jtfDate.setText(dtm.getValueAt(row, 2).toString());**

**jtfTime.setText(dtm.getValueAt(row, 3).toString());**

**jtfprice.setText(dtm.getValueAt(row, 4).toString());**

**jtfaddress.setText(dtm.getValueAt(row, 5).toString());**

**}**

**});**

**frame.setResizable(false);**

**frame.setLocationRelativeTo(null);**

**frame.setLayout(null);**

**frame.setVisible(true);**

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

**}**

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

**OrderSystem ordersystem = new OrderSystem();**

**new Login();**

**}**

**void readData() {**

**try {**

**File file = new File("data.txt");**

**file.createNewFile();**

**FileReader f = new FileReader("data.txt");**

**StringBuffer sb = new StringBuffer();**

**while (f.ready()) {**

**char c = (char) f.read();**

**if (c == '-') {**

**System.out.println(sb);**

**String dataArray[] = sb.toString().split(",");**

**Order k = new Order(dataArray[0], dataArray[1], dataArray[2], dataArray[3], dataArray[4], dataArray[5]);**

**orderlist.add(k);**

**sb = new StringBuffer();**

**} else {**

**sb.append(c);**

**}**

**}**

**dtm.setRowCount(0);**

**for (int i = 0; i < orderlist.size(); i++) {**

**Object[] objs = {orderlist.get(i).getID(), orderlist.get(i).getType(), orderlist.get(i).getDate(), orderlist.get(i).getTime(), orderlist.get(i).getPrice(), orderlist.get(i).getAddress()};**

**dtm.addRow(objs);**

**}**

**} catch (IOException e) {**

**}**

**}**

**private void writeData() {**

**try (FileWriter f = new FileWriter("data.txt")) {**

**StringBuilder sb = new StringBuilder();**

**for (int i = 0; i < orderlist.size(); i++) {**

**sb.append(orderlist.get(i).getID() + "," + orderlist.get(i).getType() + "," + orderlist.get(i).getDate() + "," + orderlist.get(i).getTime() + "," + orderlist.get(i).getPrice() + "," + orderlist.get(i).getAddress() + "-");**

**}**

**f.write(sb.toString());**

**f.close();**

**} catch (IOException e) {**

**return;**

**}**

**dtm.setRowCount(0);**

**for (int i = 0; i < orderlist.size(); i++) {**

**Object[] objs = {orderlist.get(i).getID(), orderlist.get(i).getType(), orderlist.get(i).getDate(), orderlist.get(i).getTime(), orderlist.get(i).getPrice(), orderlist.get(i).getAddress()};**

**dtm.addRow(objs);**

**}**

**}**

**public boolean isInteger(String str) {**

**if (str == null) {**

**return false;**

**}**

**int length = str.length();**

**if (length == 0) {**

**return false;**

**}**

**int i = 0;**

**if (str.charAt(0) == '-') {**

**if (length == 1) {**

**return false;**

**}**

**i = 1;**

**}**

**for (; i < length; i++) {**

**char c = str.charAt(i);**

**if (c < '0' || c > '9') {**

**return false;**

**}**

**}**

**return true;**

**}**

**}**

**JAVA CLASS ORDER CODE:**

**package ordersystem;**

**/\*\***

**\***

**\* @author User**

**\*/**

**public class Order {**

**private final String iD, type, price, address;**

**private String date, time;**

**Order(String iD, String type, String date, String time, String price, String address) {**

**this.iD = iD;**

**this.type = type;**

**this.date = date;**

**this.time = time;**

**this.price = price;**

**this.address = address;**

**}**

**String getID() {**

**return iD;**

**}**

**String getType() {**

**return type;**

**}**

**String getDate() {**

**return date;**

**}**

**String getTime() {**

**return time;**

**}**

**String getPrice() {**

**return price;**

**}**

**String getAddress() {**

**return address;**

**}**

**void setDate(String date) {**

**this.date = date;**

**}**

**void setTime(String time) {**

**this.time = time;**

**}**

**}**

**LOGIN CODE:**

**package ordersystem;**

**import javax.swing.\*;**

**import java.awt.\*;**

**import java.awt.event.\*;**

**import java.util.\*;**

**public class Login extends JFrame {**

**JButton blogin = new JButton("Login");**

**JPanel panel = new JPanel();**

**JTextField txuser = new JTextField(15);**

**JPasswordField pass = new JPasswordField(15);**

**Login() {**

**super("Login");**

**setSize(500, 400);**

**setLocation(600, 280);**

**panel.setLayout(null);**

**JLabel textuser = new JLabel("name :");**

**textuser.setBounds(45, 28, 85, 20);**

**panel.add(textuser);**

**JLabel password = new JLabel("password :");**

**password.setBounds(20, 63, 70, 20);**

**panel.add(password);**

**txuser.setBounds(100, 30, 250, 20);**

**pass.setBounds(100, 65, 250, 20);**

**blogin.setBounds(110, 100, 100, 20);**

**panel.add(blogin);**

**panel.add(txuser);**

**panel.add(pass);**

**getContentPane().add(panel);**

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

**setVisible(true);**

**setResizable(false);**

**actionlogin();**

**}**

**public void actionlogin() {**

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

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

**String puname = txuser.getText();**

**String ppaswd = pass.getText();**

**if (puname.equals("irwan") && ppaswd.equals("123456")) {**

**Login regFace = new Login();**

**regFace.setVisible(false);**

**dispose();**

**} else {**

**JOptionPane.showMessageDialog(null, "Wrong Password / Username");**

**txuser.setText("");**

**pass.setText("");**

**txuser.requestFocus();**

**}**

**}**

**});**

**}**

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

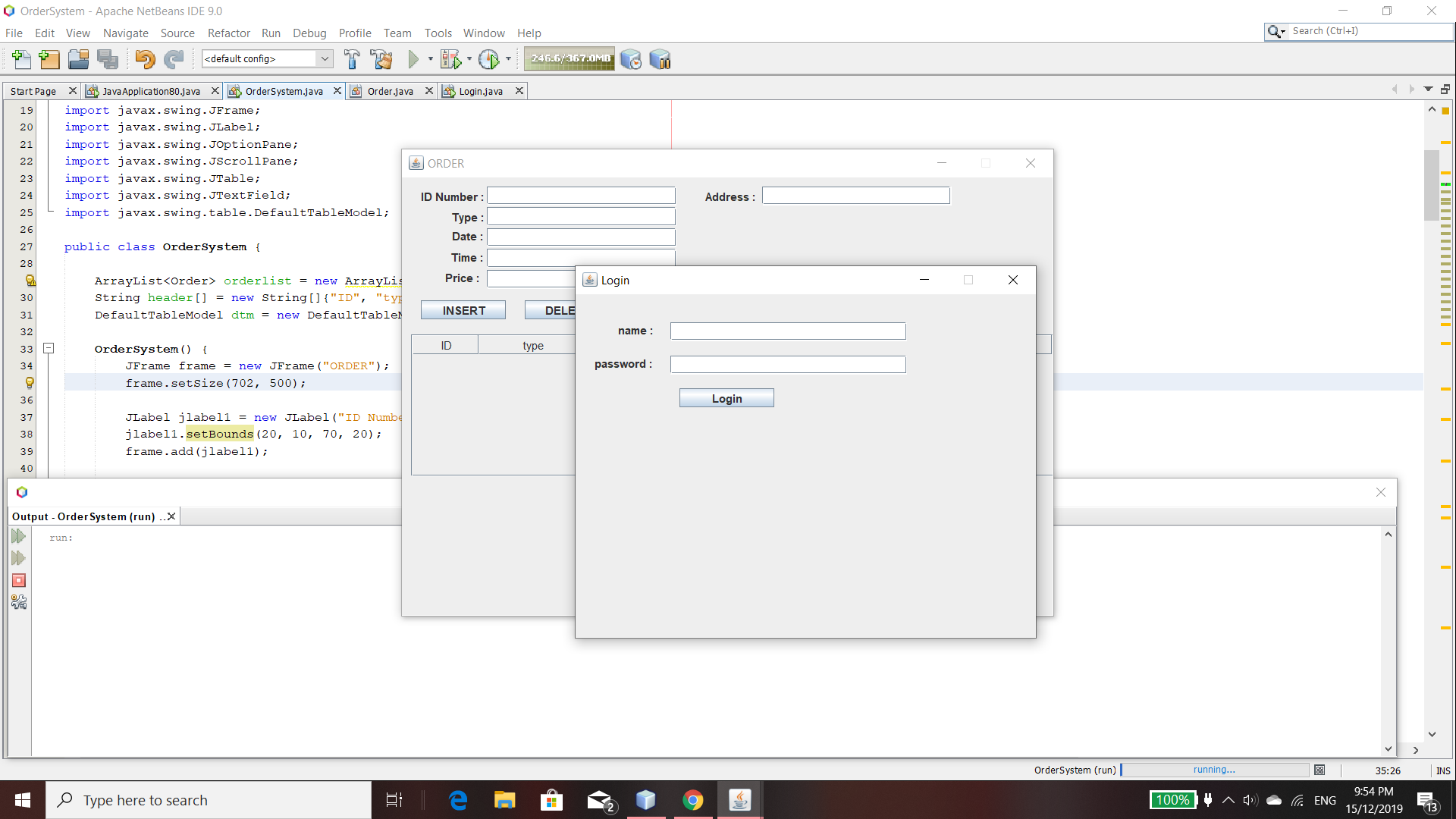
**Login login = new Login();**

**}**

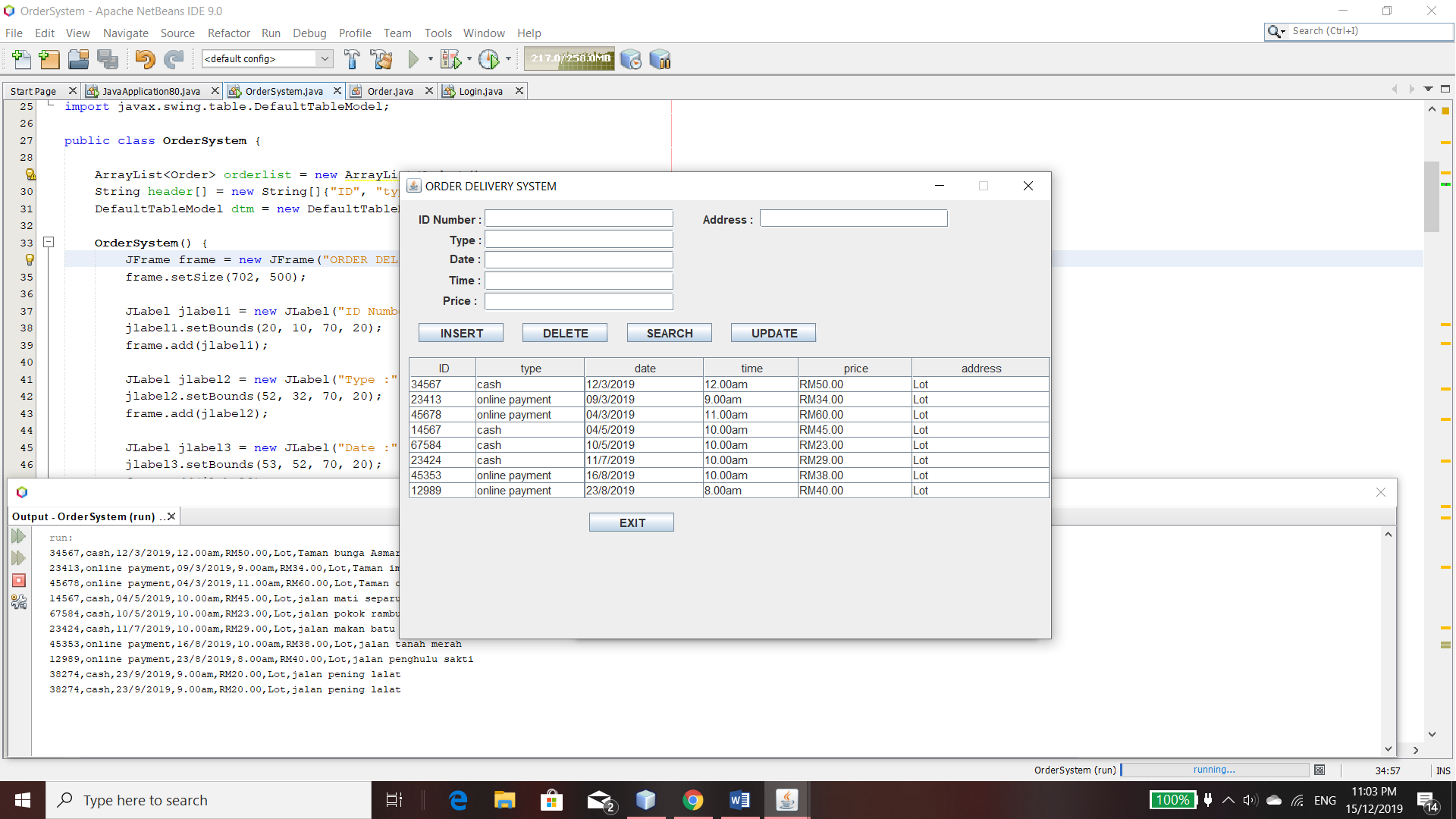
**}**

**GUI INTERFACE LOGIN AND MAIN SYSTEM:**

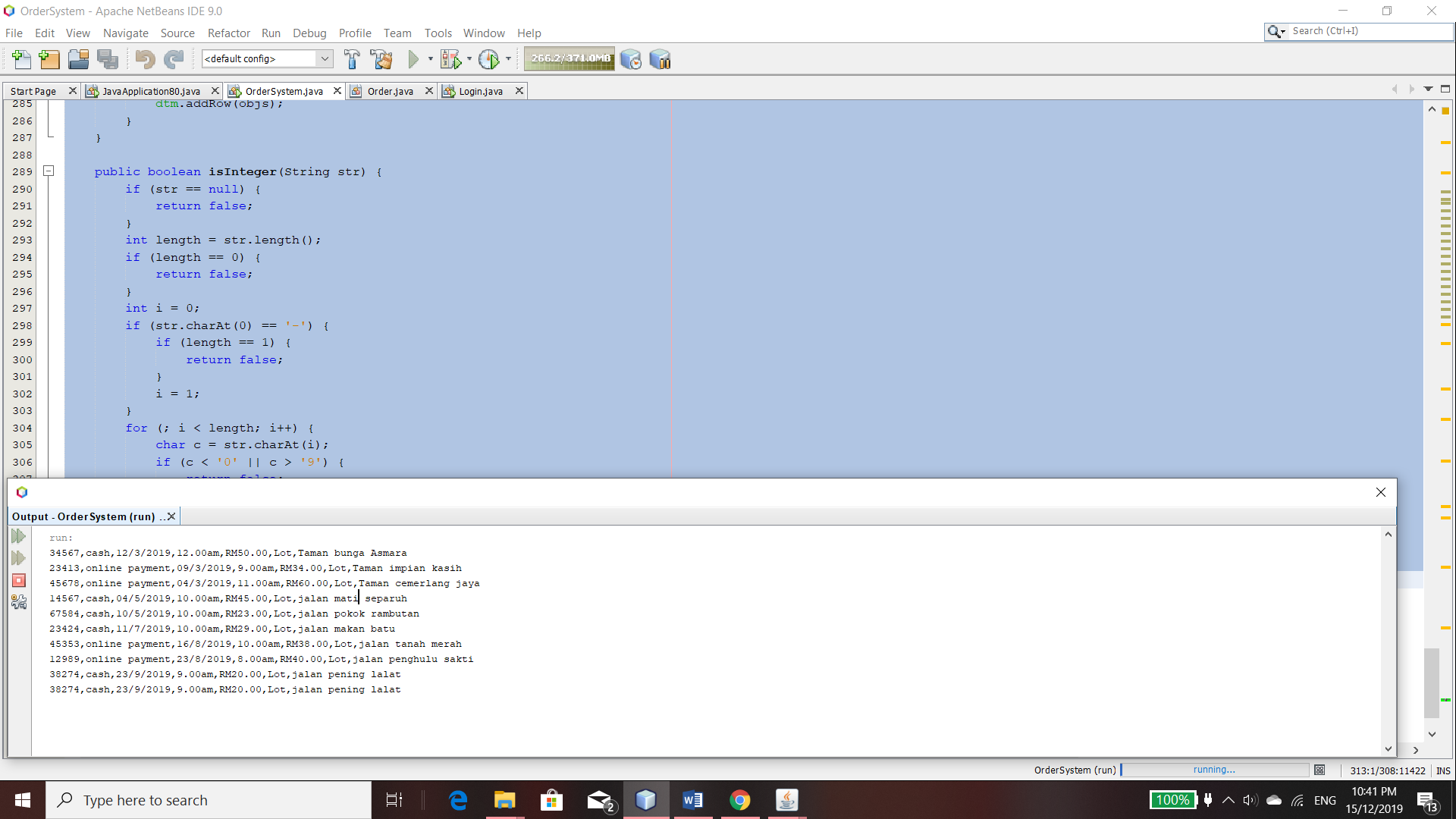
**LOGIN**

****

**MAIN SYSTEM:**



**DATABASE DATA:**

****

