2018년도

컴퓨터정보과 특성화 사업

**자바 프로젝트 진행 결과 소스코드**

**\_ 부록 \_**

(분야 : 윈도우 응용프로그램)

연구과제명 : MACAMACA 윈도우 응용프로그램 개발

2018. 12. 19



|  |
| --- |
| **본 결과물은 인하공업전문대학 컴퓨터정보과 자바 프로젝트 수업의 연구 결과 소스코드 부록 입니다.** |

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

/\*\*

\*

\* @author ChaeYoung

\*/

import java.awt.Color;

import java.awt.event.KeyEvent;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import java.util.Calendar;

import java.text.SimpleDateFormat;

import java.util.Random;

publicclass MainFrame extends javax.swing.JFrame {

DB\_MAN DBM =new DB\_MAN();

String strSQL =null;

DefaultTableModel stock\_t;

DefaultTableModel customer\_t;

DefaultTableModel orderlist\_t;

DefaultTableModel mng\_order2\_t;

DefaultTableModel mng\_order\_t;

DefaultTableModel my\_order\_t;

SimpleDateFormat sdf =new SimpleDateFormat("yyyy-MM-dd");

Calendar c1 = Calendar.getInstance();

String strToday = sdf.format(c1.getTime());

String strTime =new java.text.SimpleDateFormat("HH:mm:ss").format(new java.util.Date());

/\*\*

\* Creates new form MainFrame

\*/

public MainFrame() {

initComponents();

stock\_t = (DefaultTableModel)stock\_table.getModel();

customer\_t = (DefaultTableModel)customer\_table.getModel();

orderlist\_t = (DefaultTableModel)orderlist\_table.getModel();

mng\_order2\_t = (DefaultTableModel)MngOrder\_Order2\_table.getModel();

mng\_order\_t = (DefaultTableModel)MngOrder\_Order\_table.getModel();

my\_order\_t = (DefaultTableModel)Cus\_Order\_table.getModel();

}

String mng\_code=null;

String cus\_ID =null;

String[] cus\_order\_array = {"maca\_name","order\_count","order\_price","maca\_id"};

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

privatevoid initComponents() {

-- initComponent 제외 --

}// </editor-fold>

privatevoid btnMngActionPerformed(java.awt.event.ActionEvent evt) {

dialMngConfirm.setLocation(300,300);

dialMngConfirm.setSize(400,250);

dialMngConfirm.setVisible(true);

}

privatevoid mng\_btnStockActionPerformed(java.awt.event.ActionEvent evt) {

stock\_t.setNumRows(0);

//재고 DB 로드

try{

DBM.dbOpen();

//JOptionPane.showMessageDialog(null, "DB연결");

DBM.DB\_rs = DBM.DB\_stmt.executeQuery("Select \* from stock;");

String id=null;

String name =null;

String max =null;

String current =null;

String price =null;

while(DBM.DB\_rs.next()){

id = DBM.DB\_rs.getString(1).toString();

name = DBM.DB\_rs.getString(2).toString();

max = DBM.DB\_rs.getString(3).toString();

current = DBM.DB\_rs.getString(4).toString();

price = DBM.DB\_rs.getString(5).toString();

stock\_t.insertRow(stock\_t.getRowCount(), new Object[]{id,name,max,current,price});

}

DBM.dbClose();

//JOptionPane.showMessageDialog(null, "DB닫음");

}catch(Exception e){

JOptionPane.showMessageDialog(null, "SQLException : "+ e.getMessage());

System.out.println("SQLException : "+ e.getMessage());

}

dialMngStock.setLocation(300,300);

dialMngStock.setSize(410,500);

dialMngStock.setVisible(true);

}

privatevoid mng\_btnCustActionPerformed(java.awt.event.ActionEvent evt) {

//고객 DB 로드

customer\_t.setNumRows(0);

try{

DBM.dbOpen();

//JOptionPane.showMessageDialog(null, "DB연결");

DBM.DB\_rs = DBM.DB\_stmt.executeQuery("Select \* from customer;");

String id=null;

String pw =null;

String name =null;

String hintQ =null;

String hintA =null;

String email =null;

String vip =null;

while(DBM.DB\_rs.next()){

id = DBM.DB\_rs.getString(1).toString();

pw = DBM.DB\_rs.getString(2).toString();

name = DBM.DB\_rs.getString(3).toString();

hintQ = DBM.DB\_rs.getString(4).toString();

hintA = DBM.DB\_rs.getString(5).toString();

email = DBM.DB\_rs.getString(6).toString();

vip = DBM.DB\_rs.getString(7).toString();

customer\_t.insertRow(customer\_t.getRowCount(), new Object[]{id,name});

}

DBM.dbClose();

//JOptionPane.showMessageDialog(null, "DB닫음");

}catch(Exception e){

JOptionPane.showMessageDialog(null, "SQLException : "+ e.getMessage());

System.out.println("SQLException : "+ e.getMessage());

}

dialMngCustomer.setLocation(300,300);

dialMngCustomer.setSize(550,300);

dialMngCustomer.setVisible(true);

}

privatevoid btnCutmActionPerformed(java.awt.event.ActionEvent evt) {

frmSignIn.setLocation(300,300);

frmSignIn.setSize(450,320);

frmSignIn.setVisible(true);

}

privatevoid cus\_lblSingUpMouseClicked(java.awt.event.MouseEvent evt) {

frmSignUp.setLocation(300,300);

frmSignUp.setSize(530,600);

frmSignUp.setVisible(true);

frmSignIn.setVisible(false);

}

privatevoid cus\_SignUp\_tbxIdKeyTyped(java.awt.event.KeyEvent evt) {

cus\_SignUp\_lblIDConfirm.setVisible(true);

}

privatevoid cus\_lblFindIdMouseClicked(java.awt.event.MouseEvent evt) {

dialFindID.setLocation(300,300);

dialFindID.setSize(450,250);

dialFindID.setVisible(true);

}

privatevoid cus\_lblFindPWMouseClicked(java.awt.event.MouseEvent evt) {

dialFindPW.setLocation(300,300);

dialFindPW.setSize(450,300);

dialFindPW.setVisible(true);

}

privatevoid cus\_FPW\_btnConfirmActionPerformed(java.awt.event.ActionEvent evt) {

boolean name = cus\_FPW\_tbxName.getText().equalsIgnoreCase("") || cus\_FPW\_tbxName.getText() ==null;

boolean id = cus\_FPW\_tbxID.getText().equalsIgnoreCase("") || cus\_FPW\_tbxID.getText() ==null;

if(name || id){

JOptionPane.showMessageDialog(null, "모든정보를입력해주세요.");

}else{

try{

DBM.dbOpen();

strSQL ="select \* from customer ";

strSQL +="where name ='"+ cus\_FPW\_tbxName.getText() +"' and ";

strSQL +="cutm\_id = '"+cus\_FPW\_tbxID.getText()+"'";

System.out.println(strSQL);

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

if(DBM.DB\_rs.next()){

cus\_FPW\_tbxID.setText("");

cus\_FPW\_tbxName.setText("");

cus\_FPWq\_lblQ.setText(DBM.DB\_rs.getString("hint\_Q"));

newPW\_id.setText(DBM.DB\_rs.getString("cutm\_id"));

newPW\_name.setText(DBM.DB\_rs.getString("name"));

// 비밀번호변경창열기

newPW\_id.setVisible(false);

newPW\_name.setVisible(false);

dialFindPW.setVisible(false);

dialFindPWq.setLocation(300,300);

dialFindPWq.setSize(450,300);

dialFindPWq.setVisible(true);

cus\_FPWq\_tbxA.requestFocus();

}else{

JOptionPane.showMessageDialog(null, "이름또는아이디가잘못되었습니다.");

cus\_FPW\_tbxName.setText("");

cus\_FPW\_tbxName.requestFocus();

cus\_FPW\_tbxID.setText("");

}

DBM.dbClose();

}catch(Exception e){

System.out.println("SQLException : "+ e.getMessage());

}

}

}

privatevoid mng\_btnOrderActionPerformed(java.awt.event.ActionEvent evt) {

mng\_order\_t.setNumRows(0);

mng\_order2\_t.setNumRows(0);

try{

DBM.dbOpen();

//JOptionPane.showMessageDialog(null, "DB연결");

DBM.DB\_rs = DBM.DB\_stmt.executeQuery("SELECT order.cutm\_id, customer.name, order.maca\_id, stock.maca\_name, order.cout, order.pickup\_date, order.confirm, order.price, order.order\_date FROM (customer INNER JOIN [order] ON customer.cutm\_id = order.cutm\_id) INNER JOIN stock ON order.maca\_id = stock.maca\_id;");

String pr\_name =null;

String cout =null;

String id=null;

String name =null;

String od\_date =null;

String pic\_date =null;

String cf =null;

while(DBM.DB\_rs.next()){

pr\_name = DBM.DB\_rs.getString("maca\_name").toString();

cout = DBM.DB\_rs.getString("cout").toString();

id = DBM.DB\_rs.getString("cutm\_id").toString();

name = DBM.DB\_rs.getString("name").toString();

od\_date = DBM.DB\_rs.getString("order\_date").toString();

pic\_date = DBM.DB\_rs.getString("pickup\_date").toString();

cf = DBM.DB\_rs.getString("confirm").toString();

if(cf.equalsIgnoreCase("1")){cf ="YES";}else{cf ="NO";};

mng\_order2\_t.insertRow(mng\_order2\_t.getRowCount(), new Object[]{id,name,od\_date,pic\_date,cf});

mng\_order\_t.insertRow(mng\_order\_t.getRowCount(), new Object[]{pr\_name,cout,od\_date});

}

DBM.dbClose();

//JOptionPane.showMessageDialog(null, "DB닫음");

}catch(Exception e){

JOptionPane.showMessageDialog(null, "SQLException : "+ e.getMessage());

System.out.println("SQLException : "+ e.getMessage());

}

dialMngOrder.setLocation(300,300);

dialMngOrder.setSize(550,450);

dialMngOrder.setVisible(true);

}

privatevoid mng\_Confirm\_btnCinfirmActionPerformed(java.awt.event.ActionEvent evt) {

if(mng\_Confirm\_tbxConfirm.getText().equals("A123")) {

mng\_code = mng\_Confirm\_tbxConfirm.getText();

frmManager.setLocation(300,300);

frmManager.setSize(400,380);

frmManager.setVisible(true);

jLabel59.setText("관리자 ("+ mng\_code +") 로접속하셨습니다.");

jLabel66.setText(mng\_code+" 관리자");

jLabel67.setText(mng\_code+" 관리자 \_ 고객정보는관리자가수정할수없습니다.");

dialMngConfirm.setVisible(false);

} else {

mng\_Confirm\_lblErr.setText("관리자코드를확인해주세요");

}

}

privatevoid cus\_btnConfirmActionPerformed(java.awt.event.ActionEvent evt) {

boolean ID\_check = cus\_tbxID.getText().equalsIgnoreCase("") || cus\_tbxID ==null;

boolean PW\_check = cus\_tbxPW.getText().equalsIgnoreCase("") || cus\_tbxPW ==null;

String id = cus\_tbxID.getText();

String pw = cus\_tbxPW.getText();

if(ID\_check || PW\_check){

JOptionPane.showMessageDialog(null, "ID와 PW를모두입력하세요.");

}else{

//로그인부분

try{

DBM.dbOpen();

strSQL ="select cutm\_id, pw from customer where cutm\_id = '"+ id +"' and pw = '"+ pw +"';";

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

if(DBM.DB\_rs.next()){

frmCustomer.setLocation(300,300);

frmCustomer.setSize(400,330);

frmCustomer.setVisible(true);

frmSignIn.setVisible(false);

cus\_ID = cus\_tbxID.getText();

cus\_tbxID.setText("");

cus\_tbxPW.setText("");

} else {

JOptionPane.showMessageDialog(null, "ID 혹은 PW 오류입니다.");

}

DBM.DB\_rs.close();

DBM.dbClose();

}catch(Exception e){

System.out.println("SQLException : "+ e.getMessage());

}

}

}

privatevoid mng\_Confirm\_tbxConfirmKeyTyped(java.awt.event.KeyEvent evt) {

mng\_Confirm\_lblErr.setText("");

if(evt.getKeyChar() == KeyEvent.VK\_ENTER) {

if(mng\_Confirm\_tbxConfirm.getText().equals("A123")) {

mng\_code = mng\_Confirm\_tbxConfirm.getText();

frmManager.setLocation(300,300);

frmManager.setSize(400,380);

frmManager.setVisible(true);

jLabel59.setText("관리자 ("+ mng\_code +") 로접속하셨습니다.");

jLabel66.setText(mng\_code+" 관리자");

jLabel67.setText(mng\_code+" 관리자 \_ 고객정보는관리자가수정할수없습니다.");

dialMngConfirm.setVisible(false);

} else {

mng\_Confirm\_lblErr.setText("관리자코드를확인해주세요");

}

}

}

privatevoid mng\_Stock\_btnIdSearchActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String search\_ID =null;

search\_ID = JOptionPane.showInputDialog("검색할 ID를입력하세요.");

if(search\_ID !=null){

//DB데이터불러옴

strSQL ="Select \* from stock where maca\_id = '"+ search\_ID +"';";

try{

DBM.dbOpen();

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

while(DBM.DB\_rs.next()){

mng\_Stock\_tbxID.setText(DBM.DB\_rs.getString("maca\_id"));

mng\_Stock\_tbxName.setText(DBM.DB\_rs.getString("maca\_name"));

mng\_Stock\_tbxPrice.setText(DBM.DB\_rs.getString("price"));

mng\_Stock\_tbxMax.setText(DBM.DB\_rs.getString("max\_stock"));

mng\_Stock\_tbxCurrent.setText(DBM.DB\_rs.getString("current\_stock"));

}

DBM.DB\_rs.close();

DBM.dbClose();

}catch(Exception e){

System.out.println("SQLException : "+ e.getMessage());

}

//mng\_Stock\_tbxID.setText(search\_ID);

search\_ID =null;

}

}

privatevoid mng\_Stock\_btnNameSearchActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String search\_Name =null;

search\_Name = (JOptionPane.showInputDialog("검색할이름을입력하세요.")).toUpperCase();

if(search\_Name !=null){

strSQL ="Select \* from stock where maca\_name = '"+ search\_Name +"';";

try{

DBM.dbOpen();

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

while(DBM.DB\_rs.next()){

mng\_Stock\_tbxID.setText(DBM.DB\_rs.getString("maca\_id"));

mng\_Stock\_tbxName.setText(DBM.DB\_rs.getString("maca\_name"));

mng\_Stock\_tbxPrice.setText(DBM.DB\_rs.getString("price"));

mng\_Stock\_tbxMax.setText(DBM.DB\_rs.getString("max\_stock"));

mng\_Stock\_tbxCurrent.setText(DBM.DB\_rs.getString("current\_stock"));

}

DBM.DB\_rs.close();

DBM.dbClose();

}catch(Exception e){

System.out.println("SQLException : "+ e.getMessage());

}

search\_Name =null;

}

}

boolean check\_Modify =false;

privatevoid mng\_Stock\_btnModifyActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(!check\_Modify){ //수정활성화

check\_Modify =true;

mng\_Stock\_btnModify.setText("수정완료");

mng\_Stock\_tbxPrice.setFocusable(true);

mng\_Stock\_tbxName.setFocusable(true);

mng\_Stock\_tbxMax.setFocusable(true);

}else{ // DB update - 수정하기.

boolean tbxName = mng\_Stock\_tbxName.getText() ==null|| mng\_Stock\_tbxName.getText().equalsIgnoreCase("");

boolean tbxMax = mng\_Stock\_tbxMax.getText() ==null|| mng\_Stock\_tbxMax.getText().equalsIgnoreCase("");

boolean tbxPrice = mng\_Stock\_tbxPrice.getText() ==null|| mng\_Stock\_tbxPrice.getText().equalsIgnoreCase("");

if(tbxName || tbxMax || tbxPrice){

JOptionPane.showMessageDialog(null, "수정할모든사항을입력하여주세요.");

}else{

//DB 데이터수정

strSQL ="update stock set ";

strSQL +="maca\_name = '"+ mng\_Stock\_tbxName.getText() +"', ";

strSQL +="price = "+ mng\_Stock\_tbxPrice.getText()+", ";

strSQL +="max\_stock = "+mng\_Stock\_tbxMax.getText()+" ";

strSQL +="where maca\_id = '"+mng\_Stock\_tbxID.getText()+"';";

try{

stock\_t.setNumRows(0);

DBM.dbOpen();

DBM.DB\_stmt.executeUpdate(strSQL);

strSQL ="select \* from stock;";

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

String id=null;

String name =null;

String max =null;

String current =null;

String price =null;

while(DBM.DB\_rs.next()){

id = DBM.DB\_rs.getString(1).toString();

name = DBM.DB\_rs.getString(2).toString();

max = DBM.DB\_rs.getString(3).toString();

current = DBM.DB\_rs.getString(4).toString();

price = DBM.DB\_rs.getString(5).toString();

stock\_t.insertRow(stock\_t.getRowCount(), new Object[]{id,name,max,current,price});

}

DBM.dbClose();

}catch(Exception e){

System.out.println("SQLException : "+ e.getMessage());

}

JOptionPane.showMessageDialog(null, "수정되었습니다.");

mng\_Stock\_tbxName.setFocusable(false);

mng\_Stock\_tbxPrice.setFocusable(false);

mng\_Stock\_tbxMax.setFocusable(false);

}

check\_Modify =false;

mng\_Stock\_btnModify.setText("수정하기");

}

}

privatevoid cus\_Order\_btnSearchActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

boolean tbxID = cus\_Order\_tbxID.getText().equalsIgnoreCase("") || cus\_Order\_tbxID ==null;

boolean tbxName = cus\_Order\_tbxName.getText().equalsIgnoreCase("") || cus\_Order\_tbxName ==null;

String sql ="SELECT order.cutm\_id, customer.name, order.maca\_id, stock.maca\_name, order.cout, order.pickup\_date, order.confirm, order.price, order.order\_date FROM (customer INNER JOIN [order] ON customer.cutm\_id = order.cutm\_id) INNER JOIN stock ON order.maca\_id = stock.maca\_id ";

if( tbxID && tbxName){ //동명이인검색허락.

JOptionPane.showMessageDialog(null, "ID와이름중하나는꼭입력해주십시오.");

}elseif(!tbxID &&!tbxName){ //모두입력시

sql +="where customer.name = '"+ cus\_Order\_tbxName.getText() +"' and order.cutm\_id = '"+ cus\_Order\_tbxID.getText() +"'";

}elseif(!tbxID){ //고객 ID로만찾기

sql +="where order.cutm\_id = '"+ cus\_Order\_tbxID.getText() +"'";

}elseif(!tbxName){ //고객 name으로만찾기

sql +="where customer.name = '"+ cus\_Order\_tbxName.getText() +"'";

}

mng\_order\_t.setNumRows(0);

mng\_order2\_t.setNumRows(0);

try{

DBM.dbOpen();

//JOptionPane.showMessageDialog(null, "DB연결");

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(sql);

String pr\_name =null;

String cout =null;

String id=null;

String name =null;

String od\_date =null;

String pic\_date =null;

String cf =null;

while(DBM.DB\_rs.next()){

pr\_name = DBM.DB\_rs.getString("maca\_name").toString();

cout = DBM.DB\_rs.getString("cout").toString();

id = DBM.DB\_rs.getString("cutm\_id").toString();

name = DBM.DB\_rs.getString("name").toString();

od\_date = DBM.DB\_rs.getString("order\_date").toString();

pic\_date = DBM.DB\_rs.getString("pickup\_date").toString();

cf = DBM.DB\_rs.getString("confirm").toString();

if(cf.equalsIgnoreCase("1")){cf ="YES";}else{cf ="NO";};

mng\_order2\_t.insertRow(mng\_order2\_t.getRowCount(), new Object[]{id,name,od\_date,pic\_date,cf});

mng\_order\_t.insertRow(mng\_order\_t.getRowCount(), new Object[]{pr\_name,cout,od\_date});

}

DBM.dbClose();

//JOptionPane.showMessageDialog(null, "DB닫음");

}catch(Exception e){

JOptionPane.showMessageDialog(null, "SQLException : "+ e.getMessage());

System.out.println("SQLException : "+ e.getMessage());

}

}

privatevoid cus\_FID\_btnConfirmActionPerformed(java.awt.event.ActionEvent evt) {

// 아이디와 Email입력값체크(name, email로구분)

boolean name = cus\_FID\_tbxName.getText().equalsIgnoreCase("") || cus\_FID\_tbxName.getText() ==null;

boolean email = cus\_FID\_tbxEmail.getText().equalsIgnoreCase("") || cus\_FID\_tbxEmail.getText() ==null;

if(name || email){

JOptionPane.showMessageDialog(null, "이름과 Email 모두입력해주십시오.");

}else{

try{

DBM.dbOpen();

strSQL ="select \* from customer ";

strSQL +="where name ='"+ cus\_FID\_tbxName.getText() +"' and ";

strSQL +="email = '"+cus\_FID\_tbxEmail.getText()+"'";

System.out.println(strSQL);

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

if(DBM.DB\_rs.next()){

JOptionPane.showMessageDialog(null, cus\_FID\_tbxName.getText() +"님의 ID는 "+ DBM.DB\_rs.getString("cutm\_id") +"입니다.");

dialFindID.setVisible(false);

frmSignIn.setVisible(true);

cus\_tbxID.setText(DBM.DB\_rs.getString("cutm\_id"));

cus\_tbxPW.setText("");

}else{

JOptionPane.showMessageDialog(null, "이름또는이메일이잘못되었습니다.");

cus\_tbxID.setText("");

cus\_tbxID.requestFocus();

cus\_tbxPW.setText("");

}

DBM.dbClose();

}catch(Exception e){

System.out.println("SQLException : "+ e.getMessage());

}

}

}

privatevoid cus\_FPWq\_btnConfirmActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(cus\_FPWq\_tbxA.getText().equalsIgnoreCase("") || cus\_FPWq\_tbxA.getText() ==null){

JOptionPane.showMessageDialog(null, "답입력바랍니다.");

cus\_FPWq\_tbxA.requestFocus();

}else{

try{

DBM.dbOpen();

strSQL ="select \* from customer ";

strSQL +="where name ='"+ newPW\_name.getText() +"' and ";

strSQL +="cutm\_id = '"+ newPW\_id.getText()+"'";

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

if(DBM.DB\_rs.next()){

if(cus\_FPWq\_tbxA.getText().equalsIgnoreCase(DBM.DB\_rs.getString("hint\_A"))){

dialFindPWq.setVisible(false);

// 비밀번호변경창띄우기.

dialNewPW.setLocation(300,300);

dialNewPW.setSize(450,300);

dialNewPW.setVisible(true);

tbxNewPW.setText("");

tbxNewPW2.setText("");

lbloldPW.setVisible(false);

tbxoldPW.setVisible(false);

}else{

JOptionPane.showMessageDialog(null, "비밀번호힌트답이잘못되었습니다.");

cus\_FPWq\_tbxA.setText("");

cus\_FPWq\_tbxA.requestFocus();

}

}

DBM.dbClose();

}catch(Exception e){

System.out.println("SQLException : "+ e.getMessage());

}

}

}

privatevoid btnOkNewPwActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

try{

DBM.dbOpen();

String pw = tbxoldPW.getText();

strSQL ="select pw from customer where cutm\_id = '"+cus\_ID +"' and pw = '"+ pw +"';";

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

System.out.println(strSQL);

if(DBM.DB\_rs.next()){

//비밀번호존재시 pass

} else {

JOptionPane.showMessageDialog(null,"비밀번호를확인해주세요");

return;

}

DBM.dbClose();

//JOptionPane.showMessageDialog(null, "DB닫음");

}catch(Exception e){

JOptionPane.showMessageDialog(null, "SQLException : "+ e.getMessage());

System.out.println("SQLException : "+ e.getMessage());

}

//비밀번호글자수 check

if(!(tbxNewPW.getText().length() >=8&& tbxNewPW.getText().length() <=12)){

JOptionPane.showMessageDialog(null, "비밀번호는 8 ~ 12자입니다.");

tbxNewPW.setText("");

tbxNewPW.requestFocus();

return;

}

//비밀번호동일한지확인

if(lblCheck\_NewPW2.getText().equalsIgnoreCase("비밀번호불일치")){

JOptionPane.showMessageDialog(null, "비밀번호가불일치합니다.");

tbxNewPW.setText("");

tbxNewPW2.setText("");

tbxNewPW.requestFocus();

return;

}else{

//비밀번호 DB 변경

try{

DBM.dbOpen();

String pw = tbxoldPW.getText();

String new\_pw = tbxNewPW.getText();

strSQL ="update customer set pw = '"+new\_pw +"'"+"where cutm\_id = '"+cus\_ID +"' and pw = '"+ pw +"';";

DBM.DB\_stmt.executeUpdate(strSQL);

System.out.println(strSQL);

DBM.dbClose();

JOptionPane.showMessageDialog(null, "비밀번호가변경되었습니다.");

dialNewPW.setVisible(false);

frmCustomer.setVisible(false);

frmSignIn.setVisible(true);

cus\_tbxID.setText("");

cus\_tbxPW.setText("");

}catch(Exception e){

JOptionPane.showMessageDialog(null, "SQLException : "+ e.getMessage());

System.out.println("SQLException : "+ e.getMessage());

}

}

}

privatevoid cus\_Order\_btnOrderActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

boolean name = cus\_Order\_tbxPName.getText().equalsIgnoreCase("") || cus\_Order\_tbxPName.getText() ==null;

boolean count = cus\_Order\_tbxPcount.getText().equalsIgnoreCase("") || cus\_Order\_tbxPcount.getText() ==null;

if(name || count){

JOptionPane.showMessageDialog(null, "모든정보를입력해주세요.");

}else{

//주문확인다이알정보입력

cus\_order\_array[0] = cus\_Order\_tbxPName.getText();

cus\_order\_array[1] = cus\_Order\_tbxPcount.getText();

CheckOrder\_lblCustom.setText(cus\_ID);

CheckOrder\_lblProduct.setText(cus\_order\_array[0]);

CheckOrder\_lblCount.setText(cus\_order\_array[1]);

CheckOrder\_lblPrice.setText(cus\_order\_array[2]);

CheckOrder\_lblDate.setText(strToday +" "+ strTime);

c1.add(Calendar.DATE,7);

String pick\_date =new SimpleDateFormat("yyyy-MM-dd").format(c1.getTime());

CheckOrder\_lblPick.setText(pick\_date);

//주문확인다이알활성화

cus\_Order\_tbxPcount.setText("");

cus\_Order\_tbxPName.setText("");

dialOrder.setVisible(false);

dialCheckOrder.setLocation(300,300);

dialCheckOrder.setSize(400,457);

dialCheckOrder.setVisible(true);

}

}

privatevoid tbxNewPW2KeyPressed(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

}

privatevoid tbxNewPW2InputMethodTextChanged(java.awt.event.InputMethodEvent evt) {

// TODO add your handling code here:

}

privatevoid tbxNewPW2ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

privatevoid tbxNewPW2FocusLost(java.awt.event.FocusEvent evt) {

// TODO add your handling code here:

}

boolean distinct\_click =false;

privatevoid cus\_SignUp\_btnConfirmActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

boolean distinct = cus\_SignUp\_tbxId.getText() ==null|| cus\_SignUp\_tbxId.getText().equalsIgnoreCase("");

if(distinct){

JOptionPane.showMessageDialog(null, "ID를입력하여주세요.");

cus\_SignUp\_tbxId.requestFocus();

return;

}

if(!distinct\_click){

boolean check =false;

//중복인지디비와확인.

String id =cus\_SignUp\_tbxId.getText();

System.out.println(id);

try{

DBM.dbOpen();

//JOptionPane.showMessageDialog(null, "DB연결");

strSQL ="select cutm\_id from customer where cutm\_id = '";

strSQL += id +"'";

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

if(DBM.DB\_rs.next()){

check =false;

}else{check =true;}

DBM.dbClose();

//JOptionPane.showMessageDialog(null, "DB닫음");

}catch(Exception e){

JOptionPane.showMessageDialog(null, "SQLException : "+ e.getMessage());

System.out.println("SQLException : "+ e.getMessage());

}

if(check){//사용가능시

distinct\_click =true;

JOptionPane.showMessageDialog(null, "사용가능한 ID입니다.");

cus\_SignUp\_lblIDConfirm.setVisible(false);

}else{//사용불가능시

distinct\_click =false;

JOptionPane.showMessageDialog(null, "ID가중복됩니다.");

cus\_SignUp\_tbxId.setText("");

cus\_SignUp\_tbxId.requestFocus();

cus\_SignUp\_lblIDConfirm.setVisible(true);

}

}

}

privatevoid cus\_SignUp\_tbxPWConfirmFocusLost(java.awt.event.FocusEvent evt) {

// TODO add your handling code here:

}

privatevoid cus\_SignUp\_btnSignUpActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

boolean id\_check = cus\_SignUp\_tbxId.getText().equalsIgnoreCase("") || cus\_SignUp\_tbxId.getText() ==null;

boolean pw2\_check = cus\_SignUp\_tbxPWConfirm.getText().equalsIgnoreCase("") || cus\_SignUp\_tbxPWConfirm.getText() ==null;

boolean pw\_check = cus\_SignUp\_tbxPW.getText().equalsIgnoreCase("") || cus\_SignUp\_tbxPW.getText() ==null;

boolean name\_check = cus\_SignUp\_tbxName.getText().equalsIgnoreCase("") || cus\_SignUp\_tbxName.getText() ==null;

boolean q\_check = cus\_SignUp\_tbxQ.getText().equalsIgnoreCase("") || cus\_SignUp\_tbxQ.getText() ==null;

boolean a\_check = cus\_SignUp\_tbxA.getText().equalsIgnoreCase("") || cus\_SignUp\_tbxA.getText() ==null;

boolean email\_check = cus\_SignUp\_tbxEmail.getText().equalsIgnoreCase("") || cus\_SignUp\_tbxEmail.getText() ==null;

if(id\_check || pw2\_check || pw\_check || name\_check || q\_check || a\_check || email\_check){

JOptionPane.showMessageDialog(null, "모든정보를입력바랍니다.");

return;

}

if(!distinct\_click){

JOptionPane.showMessageDialog(null, "ID 중복확인바랍니다.");

return;

}

if(!(cus\_SignUp\_tbxPW.getText().length() >=8&& cus\_SignUp\_tbxPW.getText().length() <=12)){

JOptionPane.showMessageDialog(null, "비밀번호는 8 ~ 12자입니다.");

cus\_SignUp\_tbxPW.setText("");

cus\_SignUp\_tbxPWConfirm.setText("");

cus\_SignUp\_tbxPW.requestFocus();

return;

}

if(cus\_SignUp\_lblPW.getText().equalsIgnoreCase("PW 불일치")){

JOptionPane.showMessageDialog(null, "PW가불일치합니다.");

cus\_SignUp\_tbxPW.setText("");

cus\_SignUp\_tbxPWConfirm.setText("");

cus\_SignUp\_tbxPW.requestFocus();

return;

}

if(cus\_SignUp\_tbxEmail.getText().length() <6||!cus\_SignUp\_tbxEmail.getText().contains("@")){

JOptionPane.showMessageDialog(null, "E-mail형식을올바르게입력바랍니다.");

cus\_SignUp\_tbxEmail.setText("");

cus\_SignUp\_tbxEmail.requestFocus();

return;

}

//DB에입력

String id=cus\_SignUp\_tbxId.getText();

String pw = cus\_SignUp\_tbxPW.getText();

String name = cus\_SignUp\_tbxName.getText();

String hintQ = cus\_SignUp\_tbxQ.getText();

String hintA = cus\_SignUp\_tbxA.getText();

String email = cus\_SignUp\_tbxEmail.getText();

boolean vip =false;

strSQL="Insert Into customer(cutm\_id,pw,name,hint\_Q,hint\_A,email,vip) Values(";

strSQL+="'"+ id +"',";

strSQL+="'"+ pw +"',";

strSQL+="'"+ name +"',";

strSQL+="'"+ hintQ +"',";

strSQL+="'"+ hintA +"',";

strSQL+="'"+ email +"',";

strSQL+= vip +");";

System.out.println(strSQL);

try{

DBM.dbOpen();

DBM.DB\_stmt.executeUpdate(strSQL);

DBM.dbClose();

JOptionPane.showMessageDialog(null, "가입이완료되었습니다.");

} catch(Exception e){

System.out.println("SQLException :"+ e.getLocalizedMessage());

}

frmSignUp.setVisible(false);

}

privatevoid cus\_main\_btnOrderActionPerformed(java.awt.event.ActionEvent evt) {

//my\_order\_t.setRowCount(0);

try{

DBM.dbOpen();

strSQL ="select \* from stock";

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

String maca\_name =null;

String stock =null;

String price =null;

while(DBM.DB\_rs.next()){

maca\_name = DBM.DB\_rs.getString("maca\_name");

stock = DBM.DB\_rs.getString("current\_stock");

price = DBM.DB\_rs.getString("price");

my\_order\_t.insertRow(my\_order\_t.getRowCount(), new Object[]{maca\_name, stock, price});

}

DBM.dbClose();

}catch(Exception e){

System.out.println("SQLException : "+ e.getMessage());

}

// 주문하기폼

cus\_Order\_tbxPName.setText("");

cus\_Order\_tbxPcount.setText("");

dialOrder.setLocation(300,300);

dialOrder.setSize(410,370);

dialOrder.setVisible(true);

}

privatevoid cus\_main\_btnOlistActionPerformed(java.awt.event.ActionEvent evt) {

orderlist\_t.setNumRows(0);

try{

DBM.dbOpen();

DBM.DB\_rs = DBM.DB\_stmt.executeQuery("SELECT order.cutm\_id, order.maca\_id, stock.maca\_name, order.cout, order.pickup\_date\n"+

"FROM (customer INNER JOIN [order] ON customer.cutm\_id = order.cutm\_id) INNER JOIN stock ON order.maca\_id = stock.maca\_id;");

String cutm\_id =null;

String maca\_id =null;

String maca\_name =null;

String count =null;

String pickup =null;

while(DBM.DB\_rs.next()){

cutm\_id = DBM.DB\_rs.getString(1).toString();

maca\_id = DBM.DB\_rs.getString(2).toString();

maca\_name = DBM.DB\_rs.getString(3).toString();

count = DBM.DB\_rs.getString(4).toString();

pickup = DBM.DB\_rs.getString(5).toString();

if(cutm\_id.equalsIgnoreCase(cus\_ID)) {

orderlist\_t.insertRow(orderlist\_t.getRowCount(), new Object[]{maca\_id,maca\_name,count,pickup});

}

}

DBM.dbClose();

//JOptionPane.showMessageDialog(null, "DB닫음");

}catch(Exception e){

JOptionPane.showMessageDialog(null, "SQLException : "+ e.getMessage());

System.out.println("SQLException : "+ e.getMessage());

}

dialMyOrder.setLocation(300,300);

dialMyOrder.setSize(420,300);

dialMyOrder.setVisible(true);

dialMyOrder.setLocation(300,300);

dialMyOrder.setSize(420,300);

dialMyOrder.setVisible(true);

}

privatevoid cus\_main\_btnInfoActionPerformed(java.awt.event.ActionEvent evt) {

//내정보버튼클릭시내정보보이게하고 dialLog 띄우기

try{

DBM.dbOpen();

//JOptionPane.showMessageDialog(null, "DB연결");

strSQL ="Select \* from customer where cutm\_ID = '"+ cus\_ID +"';";

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

String id = cus\_ID;

String name =null;

String hintQ =null;

String hintA =null;

String email =null;

String vip =null;

if(DBM.DB\_rs.next()){

id = DBM.DB\_rs.getString(1).toString();

name = DBM.DB\_rs.getString(3).toString();

hintQ = DBM.DB\_rs.getString(4).toString();

hintA = DBM.DB\_rs.getString(5).toString();

email = DBM.DB\_rs.getString(6).toString();

vip = DBM.DB\_rs.getString(7).toString();

System.out.println(vip);

cus\_MyInfo\_lblName.setText(name);

cus\_MyInfo\_lblID.setText(id);

cus\_MyInfo\_tbxhintQ.setText(hintQ);

cus\_MyInfo\_tbxHintA.setText(hintA);

cus\_MyInfo\_tbxEmail.setText(email);

if(vip.equals("1")) {

vip ="○";

} else {

vip ="X";

}

cus\_MyInfo\_lblVIP.setText(vip);

}

DBM.dbClose();

//JOptionPane.showMessageDialog(null, "DB닫음");

}catch(Exception e){

JOptionPane.showMessageDialog(null, "SQLException : "+ e.getMessage());

System.out.println("SQLException : "+ e.getMessage());

}

dialMyInfo.setLocation(300,300);

dialMyInfo.setSize(400,380);

dialMyInfo.setVisible(true);

}

privatevoid cus\_MyInfo\_btnUpdateActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

if(cus\_MyInfo\_btnUpdate.getText().equalsIgnoreCase("정보수정")){

cus\_MyInfo\_btnUpdate.setText("수정하기");

cus\_MyInfo\_tbxhintQ.setFocusable(true);

cus\_MyInfo\_tbxHintA.setFocusable(true);

cus\_MyInfo\_tbxEmail.setFocusable(true);

}else{

//예외처리

boolean Q = cus\_MyInfo\_tbxhintQ.getText() ==null|| cus\_MyInfo\_tbxhintQ.getText().equalsIgnoreCase("");

boolean A = cus\_MyInfo\_tbxHintA.getText() ==null|| cus\_MyInfo\_tbxHintA.getText().equalsIgnoreCase("");

boolean Email = cus\_MyInfo\_tbxEmail.getText() ==null|| cus\_MyInfo\_tbxEmail.getText().equalsIgnoreCase("");

if(Q || A || Email){

JOptionPane.showMessageDialog(null, "모든정보를입력해주세요.");

return;

}

//수정완료

JOptionPane.showMessageDialog(null, "수정되었습니다.");

cus\_MyInfo\_btnUpdate.setText("정보수정");

cus\_MyInfo\_tbxhintQ.setFocusable(false);

cus\_MyInfo\_tbxHintA.setFocusable(false);

cus\_MyInfo\_tbxEmail.setFocusable(false);

}

}

privatevoid cus\_MyInfo\_btnConfirmActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

dialMyInfo.setVisible(false);

}

privatevoid cus\_SignUp\_tbxPWKeyReleased(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(cus\_SignUp\_tbxPWConfirm.getText().equalsIgnoreCase(cus\_SignUp\_tbxPW.getText())){

cus\_SignUp\_lblPW.setText("PW 일치");

cus\_SignUp\_lblPW.setForeground(Color.BLUE);

}else{

cus\_SignUp\_lblPW.setText("PW 불일치");

cus\_SignUp\_lblPW.setForeground(Color.red);

}

}

privatevoid cus\_SignUp\_tbxPWConfirmKeyReleased(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(cus\_SignUp\_tbxPWConfirm.getText().equalsIgnoreCase(cus\_SignUp\_tbxPW.getText())){

cus\_SignUp\_lblPW.setText("PW 일치");

cus\_SignUp\_lblPW.setForeground(Color.BLUE);

}else{

cus\_SignUp\_lblPW.setText("PW 불일치");

cus\_SignUp\_lblPW.setForeground(Color.red);

}

}

privatevoid tbxNewPW2KeyReleased(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(tbxNewPW2.getText().equalsIgnoreCase(tbxNewPW.getText())){

lblCheck\_NewPW2.setText("비밀번호일치");

lblCheck\_NewPW2.setForeground(Color.BLUE);

}else{

lblCheck\_NewPW2.setText("비밀번호불일치");

lblCheck\_NewPW2.setForeground(Color.red);

}

}

privatevoid tbxNewPWKeyReleased(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(tbxNewPW2.getText().equalsIgnoreCase(tbxNewPW.getText())){

lblCheck\_NewPW2.setText("비밀번호일치");

lblCheck\_NewPW2.setForeground(Color.BLUE);

}else{

lblCheck\_NewPW2.setText("비밀번호불일치");

lblCheck\_NewPW2.setForeground(Color.red);

}

}

privatevoid jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

dialNewPW.setLocation(300,300);

dialNewPW.setSize(450,300);

dialNewPW.setVisible(true);

tbxoldPW.setText("");

tbxNewPW.setText("");

tbxNewPW2.setText("");

}

privatevoid stock\_tableMouseClicked(java.awt.event.MouseEvent evt) {

// 테이블값클릭시띄워주기!!

String id = stock\_table.getValueAt(stock\_table.getSelectedRow(), 0).toString();

String name = stock\_table.getValueAt(stock\_table.getSelectedRow(), 1).toString();

String max = stock\_table.getValueAt(stock\_table.getSelectedRow(), 2).toString();

String current = stock\_table.getValueAt(stock\_table.getSelectedRow(), 3).toString();

String price = stock\_table.getValueAt(stock\_table.getSelectedRow(), 4).toString();

mng\_Stock\_tbxID.setText(id);

mng\_Stock\_tbxName.setText(name);

mng\_Stock\_tbxMax.setText(max);

mng\_Stock\_tbxCurrent.setText(current);

mng\_Stock\_tbxPrice.setText(price);

}

privatevoid mng\_Stock\_tbxPriceKeyReleased(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(mng\_Stock\_tbxPrice.getText() ==null|| mng\_Stock\_tbxPrice.getText().equalsIgnoreCase("")){return;}

try{

int test = Integer.parseInt(mng\_Stock\_tbxPrice.getText());

}catch(Exception e){

JOptionPane.showMessageDialog(null, "숫자로입력바랍니다.");

mng\_Stock\_tbxPrice.setText("");

mng\_Stock\_tbxPrice.requestFocus();

}

}

privatevoid mng\_Stock\_tbxMaxKeyReleased(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(mng\_Stock\_tbxMax.getText() ==null|| mng\_Stock\_tbxMax.getText().equalsIgnoreCase("")){return;}

try{

int test = Integer.parseInt(mng\_Stock\_tbxMax.getText());

}catch(Exception e){

JOptionPane.showMessageDialog(null, "숫자로입력바랍니다.");

mng\_Stock\_tbxMax.setText("");

mng\_Stock\_tbxMax.requestFocus();

}

}

privatevoid customer\_tableMouseClicked(java.awt.event.MouseEvent evt) {

// 테이블값클릭시띄워주기!!

String id = customer\_t.getValueAt(customer\_table.getSelectedRow(), 0).toString();

String pw =null;

String name = customer\_t.getValueAt(customer\_table.getSelectedRow(), 1).toString();

String hintQ =null;

String hintA =null;

String email =null;

String vip =null;

try{

DBM.dbOpen();

//JOptionPane.showMessageDialog(null, "DB연결");

String sql ="Select \* from customer where cutm\_id ='";

sql += id;

sql +="';";

System.out.println(sql);

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(sql);

while(DBM.DB\_rs.next()) {

pw = DBM.DB\_rs.getString(2).toString();

hintQ = DBM.DB\_rs.getString(4).toString();

hintA = DBM.DB\_rs.getString(5).toString();

email = DBM.DB\_rs.getString(6).toString();

vip = DBM.DB\_rs.getString(7).toString();

mng\_Customer\_lblID.setText(id);

mng\_Customer\_lblName.setText(name);

System.out.println(pw +"\t"+ hintQ +"\t"+ hintA +"\t"+ email+"\t"+ vip);

mng\_Customer\_lblPW.setText(pw);

mng\_Customer\_lblHintQ.setText(hintQ);

mng\_Customer\_lblHintA.setText(hintA);

mng\_Customer\_lblEmail.setText(email);

if(vip.equals("1")) {

vip ="○";

} else {

vip ="X";

}

mng\_Customer\_lblVip.setText(vip);

DBM.dbClose();

//JOptionPane.showMessageDialog(null, "DB닫음");

break;

}

}catch(Exception e){

JOptionPane.showMessageDialog(null, "SQLException : "+ e.getMessage());

System.out.println("SQLException : "+ e.getMessage());

}

}

privatevoid MngOrder\_Order2\_tableMouseClicked(java.awt.event.MouseEvent evt) {

// TODO add your handling code here:

String id = MngOrder\_Order2\_table.getValueAt(MngOrder\_Order2\_table.getSelectedRow(), 0).toString();

String name = MngOrder\_Order2\_table.getValueAt(MngOrder\_Order2\_table.getSelectedRow(), 1).toString();

cus\_Order\_tbxName.setText(name);

cus\_Order\_tbxID.setText(id );

}

privatevoid Cus\_Order\_tableMouseClicked(java.awt.event.MouseEvent evt) {

// TODO add your handling code here:

String maca\_name = Cus\_Order\_table.getValueAt(Cus\_Order\_table.getSelectedRow(),0).toString();

cus\_Order\_tbxPName.setText(maca\_name);

cus\_order\_array[2] = Cus\_Order\_table.getValueAt(Cus\_Order\_table.getSelectedRow(),2).toString();

}

privatevoid cus\_Order\_tbxPNameKeyReleased(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

}

privatevoid cus\_Order\_tbxPcountKeyReleased(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

if(cus\_Order\_tbxPcount.getText() ==null|| cus\_Order\_tbxPcount.getText().equalsIgnoreCase("")){return;}

try{

int order\_cnt = Integer.parseInt(cus\_Order\_tbxPcount.getText());

}catch(Exception e){

JOptionPane.showMessageDialog(null, "숫자로만입력바랍니다.");

cus\_Order\_tbxPcount.setText("");

cus\_Order\_tbxPcount.requestFocus();

}

}

privatevoid CheckOrder\_btnOkActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

String maca\_id =null;

int stock =0;

int price = Integer.parseInt(CheckOrder\_lblPrice.getText());

try{

DBM.dbOpen();

strSQL ="select \* from stock where maca\_name='"+ cus\_order\_array[0] +"'";

DBM.DB\_rs = DBM.DB\_stmt.executeQuery(strSQL);

if(DBM.DB\_rs.next()){

//제고감소

maca\_id = DBM.DB\_rs.getString("maca\_id");

stock = Integer.parseInt(DBM.DB\_rs.getString("current\_stock")) - Integer.parseInt(CheckOrder\_lblCount.getText());

strSQL ="update stock set current\_stock = '"+ stock +"'";

strSQL +=" where maca\_id = '"+ maca\_id +"'";

DBM.DB\_stmt.executeUpdate(strSQL);

}else{

JOptionPane.showMessageDialog(null, "주문하신마카롱정보가잘못되었습니다.");

}

//주문정보추가 (cutm\_id,maca\_id,cout,pickup\_date,price,order\_date,)

strSQL ="insert into order(cutm\_id,maca\_id,cout,pickup\_date,confirm,price,order\_date) values(";

strSQL +="'"+ cus\_ID +"',";

strSQL +="'"+ maca\_id +"',";

strSQL +="'"+ Integer.parseInt(CheckOrder\_lblCount.getText()) +"',";

strSQL +="'"+ CheckOrder\_lblPick.getText() +"',";

strSQL +=true+",";

strSQL +="'"+ price +"',";

strSQL +="'"+ CheckOrder\_lblDate.getText() +"');";

System.out.println(strSQL);

DBM.DB\_stmt.executeUpdate(strSQL);

JOptionPane.showMessageDialog(null, "주문이완료되었습니다.");

dialCheckOrder.setVisible(false);

DBM.dbClose();

}catch(Exception e){

System.out.println(e);

}

}

privatevoid CheckOrder\_btnCActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

dialCheckOrder.setVisible(false);

}

privatevoid Cus\_Order\_tableKeyTyped(java.awt.event.KeyEvent evt) {

// TODO add your handling code here:

}

/\*\*

\* @param args the command line arguments

\*/

publicstaticvoid main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(MainFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(MainFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(MainFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(MainFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

publicvoid run() {

new MainFrame().setVisible(true);

}

});

}

// Variables declaration - do not modify

-- 변수 부 제외--

// End of variables declaration

}