import java.sql.\*;

import java.time.LocalDate;

import java.time.format.DateTimeFormatter;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import java.util.Calendar.\*;

//\*\*\*DISPLAY X & Y AXICES \*\*\*

class MyMouseAdapter extends MouseAdapter

{

mywineshop newmfw1;

MyMouseAdapter(mywineshop mfw1)

{

newmfw1 = mfw1;

}

public void mousePressed(MouseEvent me)

{

newmfw1.mousex = me.getX();

newmfw1.mousey = me.getY();

newmfw1.mousemsg = "mouse down at "+newmfw1.mousex+" "+newmfw1.mousey;

newmfw1.repaint();

}

}

//\*\*\*WINDOW CLOSE \*\*\*\*

class MyWindowAdapter extends WindowAdapter

{

public void windowClosing(WindowEvent we)

{

System.exit(0);

}

}

class mywineshop extends Frame implements ActionListener

{

String mousemsg = " ";

int mousex = 0, mousey = 0;

String msg;

Label l1,l2,l3,l4;

TextField tuser,tpwd;

Button submit ,exit;

int r=167,g=213,b=237;

Color c;

mywineshop()

{

c=new Color(r,g,b);

setBackground(c);

l1=new Label("WELCOME");

l1.setFont(new Font("arial",Font.BOLD,20));

l2=new Label("PLEASE ENTER USERNAME AND PASSWORD");

l2.setFont(new Font("arial",Font.BOLD,25));

l3=new Label("USERNAME");

// l3.setForeground(Color.red);

l3.setFont(new Font("Times New Roman",Font.BOLD,12));

l4=new Label("PASSWORD");

// l4.setForeground(Color.red);

l4.setFont(new Font("Times New Roman",Font.BOLD,12));

tuser=new TextField(10);

tuser.setFont(new Font("arial",Font.BOLD,12));

tpwd=new TextField(10);

tuser.setFont(new Font("arial",Font.BOLD,12));

tpwd.setEchoChar('\*');

submit=new Button("Login");

submit.setFont(new Font("arial",Font.BOLD,12));

exit=new Button("Exit");

exit.setFont(new Font("arial",Font.BOLD,12));

add(l1);

// add(l2);

add(l3);

add(l4);

add(tuser);

add(tpwd);

add(submit);

add(exit);

submit.addActionListener(this);

exit.addActionListener(this);

addMouseListener(new MyMouseAdapter(this));

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

setLayout(new FlowLayout());

tuser.setText("admin");

tpwd.setText("admin");

}

public void paint(Graphics g)

{

g.drawString(mousemsg, mousex, mousey);

l1.setLocation(120,65);

l2.setLocation(80,267);

l3.setLocation(65,125);

l4.setLocation(65,175);

tuser.setLocation(200,125);

tpwd.setLocation(200,175);

submit.setLocation(90,250);

submit.setSize(50,20);

exit.setLocation(195,250);

exit.setSize(50,20);

}

public void actionPerformed(ActionEvent ae)

{

//mywineshop my=new mywineshop();

String uname,pwd;

String str=ae.getActionCommand();

if(str.equals("Login"))

{

uname=tuser.getText();

pwd=tpwd.getText();

if(uname.equals("admin") && pwd.equals("admin"))

{

JOptionPane.showMessageDialog (this, "LOGIN SUCCESSFUL !!!!!");

MenuFrame f=new MenuFrame();

f.resize(650,500);

f.setVisible(true);

setVisible(false);

//msg="Login SuccessFully";

//tuser.setText("");

// tpwd.setText("");

}

}

else if(str.equals("Exit"))

{

System.exit(0);

}

repaint();

}

public static void main(String args[])

{

mywineshop my=new mywineshop();

my.setSize(350,300);

my.setVisible(true);

my.setTitle("Login Screen");

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Menu Form\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class MenuFrame extends Frame implements ActionListener

{

String mousemsg = " ";

int mousex = 0, mousey = 0;

String msg;

int r = 170, g = 201, b = 204;

Color c;

MenuFrame()

{

setTitle("Main Menu");

c=new Color(r,g,b);

setBackground(c);

MenuBar mb = new MenuBar();

setMenuBar(mb);

setSize(500,500);

setLayout(new FlowLayout());

Menu master = new Menu("MASTER");

master.setFont(new Font("arial",Font.BOLD,12));

MenuItem product,stock,supplier,customer;

master.add(product = new MenuItem("Product"));

master.add(stock = new MenuItem("Stock"));

master.add(supplier = new MenuItem("Supplier"));

master.add(customer = new MenuItem("Customer"));

mb.add(master);

/\*Menu product = new Menu("Product");

product.setFont(new Font("arial",Font.BOLD,12));

MenuItem insert,view;

product.add(insert = new MenuItem("Insert Product"));

product.add(view = new MenuItem("View Product"));

mb.add(product);

Menu stock = new Menu("Stock");

stock.setFont(new Font("arial",Font.BOLD,12));

MenuItem insertst,viewst;

stock.add(insertst = new MenuItem("Insert Stock"));

stock.add(viewst = new MenuItem("View Stock"));

mb.add(stock);

Menu supplier = new Menu("Supplier");

supplier.setFont(new Font("arial",Font.BOLD,12));

MenuItem insertsupp,viewsupp;

supplier.add(insertsupp = new MenuItem("Insert Supplier"));

supplier.add(viewsupp = new MenuItem("View Supplier"));

mb.add(supplier) ;

Menu customer = new Menu("Customer");

customer.setFont(new Font("arial",Font.BOLD,12));

MenuItem insertc,viewc;

customer.add(insertc = new MenuItem("Insert Customer"));

customer.add(viewc = new MenuItem("View Customer"));

mb.add(customer);

\*/

Menu purchase = new Menu("Purchase");

purchase.setFont(new Font("arial",Font.BOLD,12));

MenuItem po,viewpo;

purchase.add(po = new MenuItem("Purchase Order"));

// purchase.add(viewpo = new MenuItem("View Purchase Order"));

mb.add(purchase);

Menu sales = new Menu("Sales");

sales.setFont(new Font("arial",Font.BOLD,12));

MenuItem receipt,viewrec;

sales.add(receipt = new MenuItem("Receipt"));

// sales.add(viewrec = new MenuItem("View Sales"));

mb.add(sales);

Menu reports = new Menu("Reports");

reports.setFont(new Font("arial",Font.BOLD,12));

MenuItem cust,st,supp,pur,sal;

reports.add(cust = new MenuItem("Customer Report"));

reports.add(st = new MenuItem("Stock Report"));

reports.add(supp = new MenuItem("Supplier Report"));

reports.add(pur = new MenuItem("Purchase Report"));

reports.add(sal = new MenuItem("Sales Report"));

mb.add(reports);

Menu help=new Menu("Help");

help.setFont(new Font("arial",Font.BOLD,12));

MenuItem about;

help.add(about=new MenuItem("About Project"));

mb.add(help);

Menu exit=new Menu("Exit");

exit.setFont(new Font("arial",Font.BOLD,12));

MenuItem ex;

exit.add(ex=new MenuItem("Exit"));

mb.add(exit);

ex.addActionListener(this);

product.addActionListener(this);

// view.addActionListener(this);

stock.addActionListener(this);

st.addActionListener(this);

supplier.addActionListener(this);

supp.addActionListener(this);

customer.addActionListener(this);

cust.addActionListener(this);

po.addActionListener(this);

pur.addActionListener(this);

receipt.addActionListener(this);

sal.addActionListener(this);

about.addActionListener(this);

addWindowListener(new WindowAdapter(){

public void windowClosing(WindowEvent e2)

{

int respo=JOptionPane.showConfirmDialog(null,"Are you Sure to exit","Wineshop",JOptionPane.YES\_NO\_OPTION);

if (respo==JOptionPane.YES\_OPTION)

{

System.exit(0);

}

}

});

}

public void actionPerformed(ActionEvent ae)

{

String str=ae.getActionCommand();

if(str.equals("Product"))

{

myproduct ip=new myproduct();

ip.setVisible(true);

}

else if(str.equals("Product Report"))

{

viewprod vp=new viewprod();

vp.setVisible(true);

}

else if(str.equals("Stock"))

{

mystock ist=new mystock();

ist.setVisible(true) ;

}

else if(str.equals("Stock Report"))

{

viewstock vst=new viewstock();

vst.setVisible(true);

}

else if(str.equals("Supplier"))

{

mysupplier isu=new mysupplier();

isu.setVisible(true);

}

else if(str.equals("Supplier Report"))

{

viewsupp vsu=new viewsupp();

vsu.setVisible(true);

}

else if(str.equals("Customer"))

{

mycustomer ic=new mycustomer();

ic.setVisible(true);

}

else if(str.equals("Customer Report"))

{

viewcust vc=new viewcust();

vc.setVisible(true);

}

else if(str.equals("Purchase Order"))

{

mypurchase ipo=new mypurchase();

ipo.setVisible(true);

}

else if(str.equals("Purchase Report"))

{

viewpo vpo=new viewpo();

vpo.setVisible(true);

}

else if(str.equals("Receipt"))

{

myreceipt r=new myreceipt();

r.setVisible(true);

}

else if(str.equals("Sales Report"))

{

viewsales vs=new viewsales();

vs.setVisible(true);

}

else if(str.equals("Customer"))

{

customerrep cr=new customerrep();

cr.setVisible(true);

}

else if(str.equals("Stock"))

{

stockrep stor=new stockrep();

stor.setVisible(true);

}

else if(str.equals("Supplier"))

{

supplierrep sur=new supplierrep();

sur.setVisible(true);

}

else if(str.equals("Purchase"))

{

purchaserep pr=new purchaserep();

pr.setVisible(true);

}

else if(str.equals("Sales"))

{

salesrep sr=new salesrep();

sr.setVisible(true);

}

else if(str.equals("About Project"))

{

help hp=new help();

hp.setVisible(false);

}

if(str.equals("Exit"))

{

int respo=JOptionPane.showConfirmDialog(null,"Are you Sure to exit","Wineshop",JOptionPane.YES\_NO\_OPTION);

if (respo==JOptionPane.YES\_OPTION)

{

System.exit(0);

}

}

}

class MyWindowAdapter extends WindowAdapter

{

public void windowClosing(WindowEvent we)

{

System.exit(0);

}

}

public void paint(Graphics g)

{

g.drawString(mousemsg, mousex, mousey);

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Insert Product\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class myproduct extends Frame implements ActionListener,ItemListener //FocusListener

{

String url = "jdbc:mysql://localhost:3306/WineS";

Connection con;

String createString;

Statement stmt;

String mousemsg=" ";

int mousex = 0, mousey = 0;

Label l1,l2,l3,l4,l5,l6,l7,l8,l9;

TextField txtpnm,txtpric,txtpid,txtqty,txttot,txtsup;

Choice siz,ptype,txtbr;

Button add,edit,update,delete,search,save,cancel;

int r = 170, g = 201, b = 204;

Color c;

myproduct()

{

setTitle("Insert Product");

c = new Color(r,g,b);

setBackground(c);

l1 = new Label("Product Code :");

l1.setFont(new Font("arial",Font.BOLD,12));

l2 = new Label("Product Name :");

l2.setFont(new Font("arial",Font.BOLD,12));

l3 = new Label("Brand :");

l3.setFont(new Font("arial",Font.BOLD,12));

l4 = new Label("Size :");

l4.setFont(new Font("arial",Font.BOLD,12));

l5 = new Label("Price :");

l5.setFont(new Font("arial",Font.BOLD,12));

l6 = new Label("Type :");

l6.setFont(new Font("arial",Font.BOLD,12));

//Quantity

l7 = new Label("Quantity :");

l7.setFont(new Font("arial",Font.BOLD,12));

l8 = new Label("Total :");

l8.setFont(new Font("arial",Font.BOLD,12));

l9 = new Label("Supplier :");

l9.setFont(new Font("arial",Font.BOLD,12));

txtqty = new TextField(5);

txtqty.setFont(new Font("arial",Font.BOLD,12));

txtsup = new TextField(5);

txtsup.setFont(new Font("arial",Font.BOLD,12));

txttot = new TextField(5);

txttot.setFont(new Font("arial",Font.BOLD,12));

txtpid = new TextField(5);

txtpid.setFont(new Font("arial",Font.BOLD,12));

//txtpid.addFocusListener (this);

txtpnm = new TextField(30);

txtpnm.setFont(new Font("arial",Font.BOLD,12));

txtbr = new Choice();

txtbr.setFont(new Font("arial",Font.BOLD,12));

txtpric= new TextField(5);

txtpric.setFont(new Font("arial",Font.BOLD,12));

siz = new Choice();

siz.setFont(new Font("arial",Font.BOLD,12));

ptype = new Choice();

ptype.setFont(new Font("arial",Font.BOLD,12));

add = new Button ("Add New");

add.setFont(new Font("arial",Font.BOLD,12));

save = new Button("Save");

save.setFont(new Font("arial",Font.BOLD,12));

update = new Button ("Update");

update.setFont(new Font("arial",Font.BOLD,12));

search = new Button("Search");

search.setFont(new Font("arial",Font.BOLD,12));

delete = new Button("Delete");

delete.setFont(new Font("arial",Font.BOLD,12));

cancel = new Button("Cancel");

cancel.setFont(new Font("arial",Font.BOLD,12));

edit = new Button("Edit");

edit.setFont(new Font("arial",Font.BOLD,12));

add(l1);

add(l9);

add(txtsup);

add(l2);

add(l3);

add(l4);

add(l5);

add(l6);

add(l7);

add(txtqty);

siz.add(" ");

add(siz);

////adding choice control////////

siz.addItem("60ml");

siz.addItem("90ml");

siz.addItem("180ml");

siz.addItem("275ml");

siz.addItem("330ml");

siz.addItem("375ml");

siz.addItem("500ml");

siz.addItem("750ml");

siz.addItem("1000ml");

siz.addItem("2000ml");

ptype.add(" ");

add(ptype);

ptype.addItem("Beer");

ptype.addItem("Whisky");

ptype.addItem("Rum");

ptype.addItem("Vodaka");

ptype.addItem("Wine");

ptype.addItem("Gin");

add(txtpid);

add(txtpnm);

add(txtbr);

txtbr.addItem("Select Brand");

txtbr.addItem("Dindori Reserve Shiraz");

txtbr.addItem("Seagram'a Nine Hills");

txtbr.addItem("Cristal");

txtbr.addItem("BlueNons");

txtbr.addItem("Mateus");

txtbr.addItem("ReUnite");

add(txtpric);

add(l8);

add(txttot);

add(add);

add(search);

add(save);

add(update);

add(delete);

add(cancel);

add(edit);

siz.addItemListener(this);

ptype.addItemListener(this);

add.addActionListener(this);

update.addActionListener(this);

search.addActionListener(this);

delete.addActionListener(this);

save.addActionListener(this);

edit.addActionListener(this);

cancel.addActionListener(this);

// addMouseListener(new MyMouseAdapter(this));

setSize(495,430);

setLayout(new FlowLayout());

txtpid.requestFocus();

siz.setEnabled(false);

ptype.setEnabled(false);

txtpid.setEditable(true);

txtpnm.setEditable(false);

//txtbr.setEditable(false);

txtpric.setEditable(false);

save.setEnabled(false);

delete.setEnabled(false);

update.setEnabled(false);

edit.setEnabled(false);

txtpric.addKeyListener (new KeyAdapter () {

public void keyTyped (KeyEvent ke)

{

char c = ke.getKeyChar ();

if (! ((Character.isDigit (c)) || (c == KeyEvent.VK\_BACK\_SPACE)))

{

getToolkit().beep ();

ke.consume ();

}

}

public void keyReleased(KeyEvent arg0)

{

int price=Integer.parseInt(txtpric.getText());//float

int qty=Integer.parseInt(txtqty.getText());//float

int tot=price\*qty;

txttot.setText(""+tot);

};

}

);

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

try {

Class.forName("com.mysql.jdbc.Driver");

} catch(java.lang.ClassNotFoundException e) {

System.err.print("ClassNotFoundException: ");

System.err.println(e.getMessage());

}

try {

con = DriverManager.getConnection(url,"root","");

stmt = con.createStatement();

} catch(SQLException ex) {

System.err.println("SQLException: " + ex.getMessage());

}

}

public void paint(Graphics g)

{

g.drawString(mousemsg, mousex, mousey);

g.setColor(Color.black);

g.drawRoundRect(27,55,440,350,20,20);

l1.setLocation(55,75);

l2.setLocation(55,120);

l6.setLocation(55,160);

l3.setLocation(55,200);

l4.setLocation(55,240);

l5.setLocation(55,280);

txtpid.setLocation(190,75);

siz.setLocation(190,240);

//Quantity

l7.setLocation(290, 240);

txtqty.setLocation(360,240);

//total

l8.setLocation(290, 280);

txttot.setLocation(350,280);

l9.setLocation(290, 75);

txtsup.setLocation(360,75);

txtpnm.setLocation(190,120);

txtbr.setLocation(190,200);

ptype.setLocation(190,160);

txtpric.setLocation(190,280);

add.setLocation(105,325);

search.setLocation(180,325);

update.setLocation(255,325);

delete.setLocation(330,325);

save.setLocation(180,365);

cancel.setLocation(255,365);

edit.setLocation(105,365);

add.setSize(60,20);

search.setSize(60,20);

update.setSize(60,20);

delete.setSize(60,20);

save.setSize(60,20);

cancel.setSize(60,20);

edit.setSize(60,20);

}

public void itemStateChanged(ItemEvent ie)

{

repaint();

}

public void actionPerformed (ActionEvent ae)

{

String proid,pname,brand,size,price,protype,qty,tot,sup;

String str=ae.getActionCommand();

if(str.equals("Cancel"))

{

setVisible(false);

}

if(str.equals("Add New"))

{

txtpid.setText("");

txtpnm.setText("");

//txtbr.setText("");

txtpric.setText("");

siz.select(0);

ptype.select(0);

save.setEnabled(true);

txtpid.requestFocus();

ptype.setEnabled(true);

siz.setEnabled(true);

txtpid.setEditable(true);

txtpnm.setEditable(true);

//txtbr.setEditable(true);

txtpric.setEditable(true);

txtsup.setText("");

}

else if (str.equals("Edit"))

{

txtpnm.requestFocus();

txtpid.setEditable(false);

txtpnm.setEditable(true);

//txtbr.setEditable(true);

txtpric.setEditable(true);

ptype.setEnabled(true);

siz.setEnabled(true);

update.setEnabled(true);

}

else if (str.equals("Save"))

{

txtpid.requestFocus ();

if (txtpid.getText().equals ("")||txtpnm.getText().equals ("")||ptype.getSelectedItem().equals ("")||siz.getSelectedItem().equals ("")||txtpric.getText().equals ("")||txtqty.getText().equals("")||txttot.getText().equals("")||txtsup.getText().equals(""))

{

JOptionPane.showMessageDialog (this, "All Fields Required");

}

else{

proid=txtpid.getText();

pname=txtpnm.getText();

protype=ptype.getSelectedItem();

brand=txtbr.getSelectedItem();

size=siz.getSelectedItem();

price=txtpric.getText();

qty=txtqty.getText();

tot=txttot.getText();

sup=txtsup.getText();

createString = "insert into product values('"+proid+"','"+pname+"','"+protype+"','"+brand+"','"+size+"','"+price+"','"+qty+"','"+tot+"','"+sup+"')";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record saved Successfully");

}

catch(SQLException ex)

{

System.err.println("SQLException "+ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not saved Successfully");

}

txtpid.setText("");

txtpnm.setText("");

ptype.select(0);

//txtbr.setText(" ");

txtpric.setText("");

siz.select(0);

txtbr.select(0);

txtsup.setText("");

txtpid.requestFocus();

ptype.setEnabled(false);

siz.setEnabled(false);

txtpid.setEditable(true);

txtpnm.setEditable(false);

// txtbr.setEditable(false);

txtpric.setEditable(false);

save.setEnabled(false);

}

}

else

if(str.equals("Delete"))

{

createString = "delete from product where pro\_code = '"+txtpid.getText()+"'";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record Deleted Successfully");

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not Deleted ");

}

txtpid.setText(" ");

txtpnm.setText(" ");

//txtbr.setText(" ");

txtpric.setText(" ");

txtsup.setText(" ");

siz.select(0);

ptype.select(0);

txtpid.setEditable(true);

siz.setEnabled(false);

ptype.setEnabled(false);

txtpid.setEditable(true);

txtpnm.setEditable(false);

// txtbr.setEditable(false);

txtpric.setEditable(false);

txtpid.requestFocus();

delete.setEnabled(false);

edit.setEnabled(false);

}

else if(str.equals("Search"))

{

txtpid.requestFocus ();

if (txtpid.getText().equals (""))

{

JOptionPane.showMessageDialog (this, "Product Code Required");

}

else{

try

{

createString = "select \* from product where pro\_code like '"+txtpid.getText()+"'";

ResultSet rs = stmt.executeQuery(createString);

//while(rs.next())

if(rs.next())

{

JOptionPane.showMessageDialog(this,"Record Searched Successfully");

txtpid.setText(""+rs.getString(1));

txtpnm.setText(""+rs.getString(2));

ptype.select(""+rs.getString(3));

txtbr.select(""+rs.getString(4));

siz.select(""+rs.getString(5));

txtpric.setText(""+rs.getInt(6));

txtqty.setText(""+rs.getString(7));

txttot.setText(""+rs.getString(8));

txtsup.setText(""+rs.getString(9));

}

else

JOptionPane.showMessageDialog(this,"Record not found");

txtpid.setEditable(true);

txtpnm.setEditable(false);

// txtbr.setEditable(false);

txtpric.setEditable(false);

siz.setEnabled(false);

ptype.setEnabled(false);

delete.setEnabled(true);

edit.setEnabled(true);

save.setEnabled(false);

txtpid.requestFocus();

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not found");

}

}

}

else if(str.equals("Update"))

{

createString = "update product set pro\_name = '"+txtpnm.getText()+"', pro\_type ='"+ptype.getSelectedItem()+"',brand ='"+txtbr.getSelectedItem()+"',size ='"+siz.getSelectedItem()+"',price = '"+txtpric.getText()+"',qty = '"+txtqty.getText() +"',tot = '"+txttot.getText()+ "',sup = '"+txtsup.getText()+"' where pro\_code = '"+txtpid.getText()+"'";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record Modified Successfully");

}

catch(SQLException ex)

{

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

JOptionPane.showMessageDialog(this,"Record not Modified ");

}

txtpid.setText("");

txtpnm.setText("");

txtbr.select(0);

txtpric.setText("");

txtsup.setText("");

siz.select(0);

ptype.select(0);

txtqty.setText("");

txttot.setText("");

txtpid.requestFocus();

txtpid.setEditable(true);

txtpnm.setEditable(false);

// txtbr.setEditable(false);

txtpric.setEditable(false);

ptype.setEnabled(false);

siz.setEnabled(false);

update.setEnabled(false);

edit.setEnabled(false);

}

repaint();

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* View Product \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class viewprod extends Frame //implements ActionListener

{

TextArea ta;

Label l1;

int r=197,g=223,b=228;

Color c;

viewprod()

{

c = new Color(r,g,b);

setBackground(c);

setSize(1000,700);

ta=new TextArea();

add(ta);

l1=new Label("All Product Details");

ta.setEditable(false);

add(l1);

l1.setFont(new Font("arial",Font.BOLD,25));

setTitle("View Product Details");

setLayout(new FlowLayout());

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement st=con.createStatement();

PreparedStatement ps;

ResultSet rs=null;

ps=con.prepareStatement("select \* from product ");

rs=ps.executeQuery();

String test = " \n \n\n\tProduct Code\t\tProduct Name\t\tType\t\tBrand\t\t Size\t\tPrice\t\tSize\t\tPrice\n\n\n ";

setFont(new Font("arial",Font.BOLD,14));

ta.setText(test);

while(rs.next())

{

String PRO=rs.getString(1);

String PNM=rs.getString(2);

String TYPE = rs.getString(3);

String BRAND = rs.getString(4);

String SIZE=rs.getString(5);

int PRIC = rs.getInt(6);

test = test+"\n\n\t"+ PRO+"\t\t"+PNM+"\t\t\t"+TYPE+"\t\t\t "+BRAND+"\t\t"+SIZE+"\t\t\t"+PRIC+"\n ";

setFont(new Font("arial",Font.BOLD,12));

}

ta.setText(test);

}

catch(Exception e)

{

System.out.println(e);

}

// addMouseListener(new MyMouseAdapter(this));

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

public void paint(Graphics g)

{

//g.drawString(mousemsg, mousex, mousey);

ta.setLocation(100,200);

ta.setSize(850,450);

l1.setLocation(400,150);

}

/\*public static void main(String args[])

{

viewprod my=new viewprod();

//my.setSize(800,700);

my.setVisible(true);

my.setTitle("View Product");

}

\*/

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Insert Stock \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class mystock extends Frame implements ActionListener,ItemListener

{

String url = "jdbc:mysql://localhost/WineS";

Connection con;

String createString;

Statement stmt;

String mousemsg=" ";

int mousex = 0, mousey = 0;

Label l1,l2,l3,l4,l5,l6,l7,l8,l9,l10,l11,l12,l13,l14,l15,l16,l17;

TextField txtpdt,txtqnt,txtpp,txttotal,txtstock,sid,br;

Choice pid,siz,df,dt,bin,co,ro;

Button total,add,update,delete,search,save,cancel,edit;

int r = 170, g = 201, b = 204;

Color c;

mystock()

{

setTitle("Insert Stock");

setSize(540,640);

c = new Color(r,g,b);

setBackground(c);

l1 = new Label("Bottle");

l1.setFont(new Font("arial",Font.BOLD,12));

l2 = new Label("Product Code :");

l2.setFont(new Font("arial",Font.BOLD,12));

l3 = new Label("Brand :");

l3.setFont(new Font("arial",Font.BOLD,12));

l4 = new Label("Purchased :");

l4.setFont(new Font("arial",Font.BOLD,12));

l5 = new Label("Quantity :");

l5.setFont(new Font("arial",Font.BOLD,12));

l6 = new Label("Purchase Price :");

l6.setFont(new Font("arial",Font.BOLD,12));

l7 = new Label("Size :");

l7.setFont(new Font("arial",Font.BOLD,12));

l8 = new Label("Recommended Cellaring");

l8.setFont(new Font("arial",Font.BOLD,12));

l9 = new Label("Drink From :");

l9.setFont(new Font("arial",Font.BOLD,12));

l10 = new Label("To :");

l10.setFont(new Font("arial",Font.BOLD,12));

l11 = new Label("Location");

l11.setFont(new Font("arial",Font.BOLD,12));

l12 = new Label("Rack/Bin :");

l12.setFont(new Font("arial",Font.BOLD,12));

l13 = new Label("Column :");

l13.setFont(new Font("arial",Font.BOLD,12));

l14 = new Label("Row :");

l14.setFont(new Font("arial",Font.BOLD,12));

l15 = new Label("Supplier ID");

l15.setFont(new Font("arial",Font.BOLD,12));

l16 = new Label("Stock ID :");

l16.setFont(new Font("arial",Font.BOLD,12));

l17 = new Label("Stock Details");

l17.setFont(new Font("arial",Font.BOLD,18));

txtstock = new TextField(7);

txtstock.setFont(new Font("arial",Font.BOLD,12));

txtpdt = new TextField(7);

txtpdt.setFont(new Font("arial",Font.BOLD,12));

txtqnt = new TextField(5);

txtqnt.setFont(new Font("arial",Font.BOLD,12));

txtpp = new TextField(10);

txtpp.setFont(new Font("arial",Font.BOLD,12));

txttotal = new TextField(10);

txttotal.setFont(new Font("arial",Font.BOLD,12));

DateTimeFormatter dtf = DateTimeFormatter.ofPattern("yyyy/MM/dd");

LocalDate localDate = LocalDate.now();

System.out.println(dtf.format(localDate));

txtpdt.setText(localDate.toString());

pid = new Choice();

pid.setFont(new Font("arial",Font.BOLD,12));

pid.addItemListener(new ItemListener()

{

@Override

public void itemStateChanged(ItemEvent arg0)

{

String s=pid.getSelectedItem();

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement stmt = conn.createStatement();

String cmdsup = "Select \* from product where pro\_code='"+s+"'";

System.out.println(cmdsup);

ResultSet rssup = stmt.executeQuery(cmdsup);

while(rssup.next())

{

sid.setText(rssup.getString(9).toString());

br.setText(rssup.getString(4));

siz.select(rssup.getString(5));

txtqnt.setText(rssup.getString(7));

txtpp.setText(rssup.getString(6));

txttotal.setText(rssup.getString(8));

}//while

stmt.close();

conn.close();

}//try

catch(Exception e)

{

System.out.println(e);

}//catch

}

});

br = new TextField(20);

br.setFont(new Font("arial",Font.BOLD,12));

siz = new Choice();

siz.setFont(new Font("arial",Font.BOLD,12));

df = new Choice();

df.setFont(new Font("arial",Font.BOLD,12));

dt = new Choice();

dt.setFont(new Font("arial",Font.BOLD,12));

bin = new Choice();

bin.setFont(new Font("arial",Font.BOLD,12));

co = new Choice();

co.setFont(new Font("arial",Font.BOLD,12));

ro = new Choice();

ro.setFont(new Font("arial",Font.BOLD,12));

sid = new TextField(5);

sid.setFont(new Font("arial",Font.BOLD,12));

total = new Button ("Total Amount");

total.setFont(new Font("arial",Font.BOLD,12));

add = new Button ("Add New");

add.setFont(new Font("arial",Font.BOLD,12));

save = new Button("Save");

save.setFont(new Font("arial",Font.BOLD,12));

edit = new Button ("Edit");

edit.setFont(new Font("arial",Font.BOLD,12));

update = new Button ("Update");

update.setFont(new Font("arial",Font.BOLD,12));

search = new Button("Search");

search.setFont(new Font("arial",Font.BOLD,12));

delete = new Button("Delete");

delete.setFont(new Font("arial",Font.BOLD,12));

cancel = new Button("Cancel");

cancel.setFont(new Font("arial",Font.BOLD,12));

add(l1);

add(l2);

add(l3);

add(l4);

add(l5);

add(l6);

add(l7);

add(l8);

add(l9);

add(l10);

add(l11);

add(l12);

add(l13);

add(l14);

add(l15);

add(l16);

add(l17);

pid.add(" ");

add(pid);

//br.add(" ");

add(br); ;

siz.add(" ");

add(siz); ;

////adding choice control////////

siz.addItem("60ml");

siz.addItem("90ml");

siz.addItem("180ml");

siz.addItem("275ml");

siz.addItem("330ml");

siz.addItem("375ml");

siz.addItem("500ml");

siz.addItem("750ml");

siz.addItem("1000ml");

siz.addItem("2000ml");

df.add(" ");

add(df);

dt.add(" ");

add(dt); ;

df.add("2009"); dt.add("2009");

df.add("2010"); dt.add("2010");

df.add("2011"); dt.add("2011");

df.add("2012"); dt.add("2012");

df.add("2013"); dt.add("2013");

df.add("2014"); dt.add("2014");

df.add("2015"); dt.add("2015");

df.add("2016"); dt.add("2016");

df.add("2017"); dt.add("2017");

df.add("2018"); dt.add("2018");

df.add("2019"); dt.add("2019");

df.add("2020"); dt.add("2020");

bin.add(" ");

add(bin); ;

////adding choice control////////

bin.addItem("Wooden Rack East 1");

bin.addItem("Wooden Rack East 2");

bin.addItem("Wooden Rack West 1");

bin.addItem("Wooden Rack West 2");

bin.addItem("Wooden Rack South 1");

bin.addItem("Wooden Rack South 2");

bin.addItem("Wooden Rack North 1");

bin.addItem("Wooden Rack North 2");

co.add(" ");

add(co); ;

co.add("A");

co.add("B");

co.add("C");

co.add("D");

co.add("E");

co.add("F");

co.add("G");

co.add("H");

co.add("I");

co.add("J");

co.add("K");

co.add("L");

co.add("M");

co.add("N");

co.add("O");

co.add("P");

ro.add(" ");

add(ro); ;

ro.add("1");

ro.add("2");

ro.add("3");

ro.add("4");

ro.add("5");

ro.add("6");

ro.add("7");

ro.add("8");

ro.add("9");

ro.add("10");

ro.add("11");

ro.add("12");

ro.add("13");

ro.add("14");

ro.add("15");

ro.add("16");

//sid.add(" ");

add(sid); ;

add(txtstock);

add(txtpdt);

add(txtqnt);

add(txtpp);

add(txttotal);

add(total);

add(add);

add(edit);

add(search);

add(update);

add(save);

add(delete);

add(cancel);

pid.addItemListener(this);

//br.addItemListener(this);

siz.addItemListener(this);

df.addItemListener(this);

dt.addItemListener(this);

bin.addItemListener(this);

co.addItemListener(this);

ro.addItemListener(this);

total.addActionListener(this);

add.addActionListener(this);

edit.addActionListener(this);

update.addActionListener(this);

search.addActionListener(this);

delete.addActionListener(this);

save.addActionListener(this);

cancel.addActionListener(this);

//addWindowListener(new MyWindowAdapter());;

//addMouseListener(new MyMouseAdapter(this));

setLayout(new FlowLayout());

txtstock.setEditable(true);

txtpdt.setEditable(false);

txtqnt.setEditable(false);

txtpp.setEditable(false);

txttotal.setEditable(false);

sid.setEnabled(false);

siz.setEnabled(false);

pid.setEnabled(false);

br.setEnabled(false);

df.setEnabled(false);

dt.setEnabled(false);

bin.setEnabled(false);

co.setEnabled(false);

ro.setEnabled(false);

save.setEnabled(false);

delete.setEnabled(false);

update.setEnabled(false);

edit.setEnabled(false);

txtqnt.addKeyListener (new KeyAdapter () {

public void keyTyped (KeyEvent ke) {

char c = ke.getKeyChar ();

if (! ((Character.isDigit (c)) || (c == KeyEvent.VK\_BACK\_SPACE))) {

getToolkit().beep ();

ke.consume ();

}

}

}

);

txtpp.addKeyListener (new KeyAdapter () {

public void keyTyped (KeyEvent ke) {

char c = ke.getKeyChar ();

if (! ((Character.isDigit (c)) || (c == KeyEvent.VK\_BACK\_SPACE))) {

getToolkit().beep ();

ke.consume ();

}

}

}

);

txttotal.addKeyListener (new KeyAdapter () {

public void keyTyped (KeyEvent ke) {

char c = ke.getKeyChar ();

if (! ((Character.isDigit (c)) || (c == KeyEvent.VK\_BACK\_SPACE))) {

getToolkit().beep ();

ke.consume ();

}

}

}

);

//////sid choice box//////////

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement stmt = conn.createStatement();

String cmdsup = "Select Distinct supp\_id from supplier";

ResultSet rssup = stmt.executeQuery(cmdsup);

while(rssup.next())

{

// sid.addItem(rssup.getString(1));

}//while

stmt.close();

conn.close();

}//try

catch(Exception e)

{

System.out.println(e);

}//catch

try

{

Class.forName("com.mysql.jdbc.Driver");

}

catch(java.lang.ClassNotFoundException e)

{

System.err.print("ClassNotFoundException: ");

System.err.println(e.getMessage());

}

try

{

con = DriverManager.getConnection(url,"root","");

stmt = con.createStatement();

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

}

//////brand choice box//////////

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement stmt = conn.createStatement();

String cmdbr = "Select Distinct brand from product";

ResultSet rsbr = stmt.executeQuery(cmdbr);

while(rsbr.next())

{

//br.addItem(rsbr.getString(1));

}//while

stmt.close();

conn.close();

}//try

catch(Exception e)

{

System.out.println(e);

}//catch

//////Product Code choice box//////////

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement stmt = conn.createStatement();

String cmdpid = "Select Distinct pro\_code from product";

ResultSet rspid = stmt.executeQuery(cmdpid);

while(rspid.next())

{

pid.addItem(rspid.getString(1));

}//while

stmt.close();

conn.close();

}//try

catch(Exception e)

{

System.out.println(e);

}//catch

//Windoe hide//

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}/////Constructor()///////

public void paint(Graphics g)

{

g.drawString(mousemsg, mousex, mousey);

g.setColor(Color.black);

g.drawRoundRect(27,90,480,230,0,0);

g.drawRoundRect(27,340,480,60,0,0);

g.drawRoundRect(27,420,480,93,0,0);

l1.setLocation(62,78);

l2.setLocation(45,160);

l3.setLocation(275,200);

l4.setLocation(275,160);

l5.setLocation(275,240);

l6.setLocation(45,280);

l7.setLocation(45,240);

l8.setLocation(62,328);

l9.setLocation(45,360);

l10.setLocation(275,360);

l11.setLocation(45,408);

l12.setLocation(45,440);

l13.setLocation(45,480);

l14.setLocation(275,480);

l15.setLocation(45,200);

l16.setLocation(45,124);

l17.setLocation(42,40);

txtstock.setLocation(150,120);

txttotal.setLocation(365,280);

total.setLocation(270,280);

pid.setLocation(150,160);

sid.setLocation(150,200);

br.setLocation(341,200);

siz.setLocation(150,240);

df.setLocation(150,360);

dt.setLocation(340,360);

bin.setLocation(150,440);

co.setLocation(150,480);

ro.setLocation(340,480);

txtpdt.setLocation(360,160);

txtqnt.setLocation(341,240);

txtpp.setLocation(150,280);

add.setLocation(105,540);

search.setLocation(180,540);

update.setLocation(255,540);

delete.setLocation(330,540);

save.setLocation(180,590);

cancel.setLocation(255,590);

edit.setLocation(105,590);

add.setSize(60,20);

search.setSize(60,20);

update.setSize(60,20);

delete.setSize(60,20);

save.setSize(60,20);

cancel.setSize(60,20);

edit.setSize(60,20);

}

public void itemStateChanged(ItemEvent ie)

{

repaint();

}

public void actionPerformed (ActionEvent ae)

{

String str=ae.getActionCommand();

if (str.equals("Total Amount"))

{

txttotal.setText("");

int total;

total = Integer.parseInt(txtqnt.getText()) \* Integer.parseInt(txtpp.getText());

String stotal = txttotal.getText();

stotal = stotal + total;

txttotal.setText(stotal);

}

if(str.equals("Add New"))

{

save.setEnabled(true);

Statement stmt1;

ResultSet rs1;

try

{

Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e)

{

System.out.println("Class not found"+e);

}

try

{

Connection conn1 = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

stmt1 =conn1.createStatement();

int temp=0;

rs1=stmt1.executeQuery("select \* from stock where stock\_id=(select max(stock\_id) from stock)");

//txtstock("");

//txtpdt.setText("");

txtpp.setText("");

txtqnt.setText("");

txttotal.setText("");

siz.select(0);

pid.select(0);

sid.setText("");

br.setText("");

df.select(0);

dt.select(0);

bin.select(0);

co.select(0);

ro.select(0);

save.setEnabled(true);

txtpdt.setEditable(true);

txtpp.setEditable(true);

txtqnt.setEditable(true);

txttotal.setEditable(true);

sid.setEnabled(true);

siz.setEnabled(true);

pid.setEnabled(true);

br.setEnabled(true);

df.setEnabled(true);

dt.setEnabled(true);

bin.setEnabled(true);

co.setEnabled(true);

ro.setEnabled(true);

save.setEnabled(true);

while(rs1.next())

{

temp=((rs1.getInt("stock\_id")) +1);

String t=Integer.toString(temp);

txtstock.setText(t);

}

}

catch(Exception e)

{

System.out.println("Driver not found"+e);

}

}

else if (str.equals("Edit"))

{ txtstock.setEditable(false);

txtpdt.setEditable(true);

txtpp.setEditable(true);

txtqnt.setEditable(true);

txttotal.setEditable(true);

sid.setEnabled(true);

siz.setEnabled(true);

pid.setEnabled(true);

br.setEnabled(true);

df.setEnabled(true);

dt.setEnabled(true);

bin.setEnabled(true);

co.setEnabled(true);

ro.setEnabled(true);

pid.requestFocus();

update.setEnabled(true);

}

String proid,pdate,Quantity,purp,Size,Brand,totalamount,dfrom,dto,rac,Co,Ro,supid,stid;

if(str.equals("Cancel"))

{

setVisible(false);

}

else if (str.equals("Save"))

{

pid.requestFocus ();

if (txtstock.getText().equals ("")||pid.getSelectedItem().equals ("")||sid.getText().equals ("")||br.getText().equals ("")||txtpdt.getText().equals ("")||txtqnt.getText().equals ("")||siz.getSelectedItem().equals ("")||txtpp.getText().equals ("")||txttotal.getText().equals ("")||df.getSelectedItem().equals ("")||dt.getSelectedItem().equals ("")||bin.getSelectedItem().equals ("")||co.getSelectedItem().equals ("")||ro.getSelectedItem().equals (""))

{

JOptionPane.showMessageDialog (this, "All Fields Required");

}

else{

stid=txtstock.getText();

proid=pid.getSelectedItem();

supid=sid.getText();

Brand=br.getText();

pdate=txtpdt.getText();

Quantity=txtqnt.getText();

Size=siz.getSelectedItem();

purp=txtpp.getText();

totalamount = txttotal.getText();

dfrom =df.getSelectedItem();

dto =dt.getSelectedItem();

rac = bin.getSelectedItem();

Co = co.getSelectedItem();

Ro = ro.getSelectedItem();

createString = "insert into stock values('"+stid+"','"+proid+"','"+supid+"','"+Brand+"','"+pdate+"','"+Quantity+"','"+Size+"','"+purp+"','"+totalamount+"','"+dfrom+"','"+dto+"','"+rac+"','"+Co+"','"+Ro+"')";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record saved Successfully");

}

catch(SQLException ex)

{

System.err.println("SQLException "+ex.getMessage());

}

txtstock.setText("");

//txtpdt.setText("");

txtpp.setText("");

txtqnt.setText("");

txttotal.setText("");

siz.select(0);

pid.select(0);

sid.setText("");

br.setText("");

df.select(0);

dt.select(0);

bin.select(0);

co.select(0);

ro.select(0);

pid.requestFocus();

txtpdt.setEditable(true);

txtpp.setEditable(false);

txtqnt.setEditable(false);

txttotal.setEditable(false);

save.setEnabled(false);

sid.setEnabled(false);

siz.setEnabled(false);

pid.setEnabled(true);

br.setEnabled(false);

df.setEnabled(false);

dt.setEnabled(false);

bin.setEnabled(false);

co.setEnabled(false);

ro.setEnabled(false);

}

}

else

if(str.equals("Delete"))

{

createString = "delete from stock where stock\_id = '"+txtstock.getText()+"'";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record Deleted Successfully");

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not Deleted ");

}

txtstock.setText("");

// txtpdt.setText("");

txtpp.setText("");

txtqnt.setText("");

txttotal.setText("");

txtpdt.setText("");

siz.select(0);

pid.select(0);

sid.setText("");

br.setText("");

df.select(0);

dt.select(0);

bin.select(0);

co.select(0);

ro.select(0);

update.setEnabled(false);

delete.setEnabled(false);

edit.setEnabled(false);

}

else if(str.equals("Search"))

{

txtstock.requestFocus ();

if (txtstock.getText().equals (""))

{

JOptionPane.showMessageDialog (this, "Stock ID Required");

}

else{

try

{

createString = "select \* from stock where stock\_id like '"+txtstock.getText()+"'";

ResultSet rs = stmt.executeQuery(createString);

if(rs.next())

{

txtstock.setText(""+rs.getInt(1));

pid.select(""+rs.getString(2));

sid.setText(""+rs.getInt(3));

br.setText(""+rs.getString(4));

txtpdt.setText(""+rs.getString(5));

txtqnt.setText(""+rs.getInt(6));

siz.select(""+rs.getString(7));

txtpp.setText(""+rs.getInt(8));

txttotal.setText(""+rs.getInt(9));

df.select(""+rs.getInt(10));

dt.select(""+rs.getInt(11));

bin.select(""+rs.getString(12));

co.select(""+rs.getString(13));

ro.select(""+rs.getInt(14));

JOptionPane.showMessageDialog(this,"Record Searched Successfully");

}

else

JOptionPane.showMessageDialog(this,"Record not found");

txtpdt.setEditable(true);

txtpdt.setEditable(false);

txtpp.setEditable(false);

txtqnt.setEditable(false);

txttotal.setEditable(false);

delete.setEnabled(true);

edit.setEnabled(true);

save.setEnabled(false);

sid.setEnabled(false);

siz.setEnabled(false);

pid.setEnabled(false);

br.setEnabled(false);

df.setEnabled(false);

dt.setEnabled(false);

bin.setEnabled(false);

co.setEnabled(false);

ro.setEnabled(false);

}

catch(SQLException ex)

{ JOptionPane.showMessageDialog(this,"Record Not Found");

System.err.println("SQLException: " + ex.getMessage());

}

}}

else if(str.equals("Update"))

{

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement st=con.createStatement();

PreparedStatement ps;

ResultSet rs;

String productid=pid.getSelectedItem();

String suppid=sid.getText();

String brands=br.getText();

String purdate=txtpdt.getText() ;

String Qty=txtqnt.getText();

String sige=siz.getSelectedItem();

String puramt=txtpp.getText();

String tamt=txttotal.getText();

String drinkf=df.getSelectedItem();

String drinkto=dt.getSelectedItem();

String rak=bin.getSelectedItem();

String colm=co.getSelectedItem();

String sro=ro.getSelectedItem();

ps=con.prepareStatement("update stock set pur\_code = ?, supp\_id = ?,brand = ?,pur\_date=?,quantity=?,size=?,pur\_price=?,total\_amt=?,drink\_from=?,drink\_to=?,rack\_bin=?,colmn = ?,r\_no=? where stock\_id ="+"'"+txtstock.getText()+"'"+" ");

//ps.setInt(1,cno);

ps.setString(1,productid);

ps.setString(2,suppid);

ps.setString(3,brands);

ps.setString(4,purdate);

ps.setString(5,Qty);

ps.setString(6,sige);

ps.setString(7,puramt);

ps.setString(8,tamt);

ps.setString(9,drinkf);

ps.setString(10,drinkto);

ps.setString(11,rak);

ps.setString(12,colm);

ps.setString(13,sro);

ps.executeUpdate();

JOptionPane.showMessageDialog(null,"Record Modified Successfully!!!!!",

"Successful",JOptionPane.INFORMATION\_MESSAGE);

txtstock.setText("");

//txtpdt.setText("");

txtpp.setText("");

txtqnt.setText("");

txttotal.setText("");

siz.select(0);

pid.select(0);

sid.setText("");

br.setText("");

df.select(0);

dt.select(0);

bin.select(0);

co.select(0);

ro.select(0);

update.setEnabled(false);

delete.setEnabled(false);

edit.setEnabled(false);

txtstock.setEditable(true);

txtpdt.setEditable(false);

txtpp.setEditable(false);

txtqnt.setEditable(false);

txttotal.setEditable(false);

sid.setEnabled(false);

siz.setEnabled(false);

pid.setEnabled(false);

br.setEnabled(false);

df.setEnabled(false);

dt.setEnabled(false);

bin.setEnabled(false);

co.setEnabled(false);

ro.setEnabled(false);

}

catch(Exception e)

{

JOptionPane.showMessageDialog(null,"Record Can Not be Updated!!!",

"Failure",JOptionPane.INFORMATION\_MESSAGE);

System.out.println(e);

}

}

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* View Stock \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class viewstock extends Frame //implements ActionListener

{

String mousemsg = " ";

int mousex = 0, mousey = 0;

Label l1,l2,l3;

TextField txtdate;

Button Viewall,home;

Choice txtpcode;

int r=197,g=223,b=228;

Color c;

TextArea ta;

viewstock()

{

c = new Color(r,g,b);

setBackground(c);

setSize(1000,700);

ta=new TextArea();

add(ta);

l1=new Label("All Stock Details");

//

l2 = new Label("DATE ");

l2.setFont(new Font("arial",Font.BOLD,12));

l3 = new Label("Product code ");

l3.setFont(new Font("arial",Font.BOLD,12));

txtdate = new TextField(10);

txtdate.setFont(new Font("arial",Font.BOLD,12));

txtpcode = new Choice();

txtpcode.setFont(new Font("arial",Font.BOLD,12));

ta.setEditable(false);

add(l1);

add(l2);

add(txtdate);

add(l3);

add(txtpcode);

//

Viewall = new Button ("Viewall");

Viewall.setFont(new Font("arial",Font.BOLD,12));

home = new Button ("home");

home.setFont(new Font("arial",Font.BOLD,12));

add(Viewall);

add(home);

//Viewall.addMouseListener((MouseListener) this);

//home.addActionListener((ActionListener) this);

l1.setFont(new Font("arial",Font.BOLD,25));

setTitle("View Stock Details");

setLayout(new FlowLayout());

//Viewall.setEnabled(false);

//home.setEnabled(false);

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement st=con.createStatement();

PreparedStatement ps;

ResultSet rs=null;

ps=con.prepareStatement("select \* from stock");

rs=ps.executeQuery();

System.out.println("data fetch");

setFont(new Font("arial",Font.BOLD,14));

String test = " \n \n\n\tStock ID\t\tDate\t\t Product Code\t\tSupplier Id\t Brand\t Quantity\t\tSize\t\tPrice\t\tTotal\t\tDrink From\t\tTo\t\tRack\t\t\t\tColumn\t\tRow\n\n\n ";

ta.setText(test);

while(rs.next())

{

String STID = rs.getString(1);

String PRO=rs.getString(2);

String SID = rs.getString(3);

String BRAND=rs.getString(4);

String DATE = rs.getString(5);

int QUNT = rs.getInt(6);

String SIZE = rs.getString(7);

int PRICE = rs.getInt(8);

int TOTAL = rs.getInt(9);

int DF = rs.getInt(10);

int DT = rs.getInt(11);

String RAC=rs.getString(12);

String CO=rs.getString(13);

int RO = rs.getInt(14);

setFont(new Font("arial",Font.BOLD,12));

test = test+"\n\n\t"+ STID+"\t\t"+DATE+"\t "+PRO+"\t\t\t"+SID+"\t\t "+BRAND+"\t "+QUNT+"\t\t"+SIZE+"\t\t"+PRICE+"\t\t "+TOTAL+"\t\t"+DF+"\t\t\t "+DT+"\t\t"+RAC+"\t\t "+CO+"\t\t"+RO+"\t\t\t "+"\n ";

}

ta.setText(test);

}

catch(Exception e)

{

System.out.println(e);

}

// addMouseListener(new MyMouseAdapter(this));

//Windoe hide//

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

public void paint(Graphics g)

{

g.drawString(mousemsg, mousex, mousey);

ta.setLocation(100,200);

ta.setSize(850,450);

l1.setLocation(400,150);

///

l2.setLocation(100, 100);

txtdate.setLocation(150,100);

l3.setLocation(300, 100);

txtpcode.setLocation(350,100);

Viewall.setLocation(500,100);

home.setLocation(700,100);

Viewall.setSize(60,20);

home.setSize(60,20);

//total

//total

}

/\* public static void main(String args[])

{

viewstock my=new viewstock();

//my.setSize(800,700);

my.setVisible(true);

my.setTitle("View Stock");

}

\*/

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Insert Supplier \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class mysupplier extends Frame implements ActionListener,ItemListener

{

String url = "jdbc:mysql://localhost/WineS";

Connection con;

String createString;

Statement stmt;

String mousemsg=" ";

int mousex = 0, mousey = 0;

Label l1,l2,l3,l4,l5,l6,l7,l8;

TextField txtsid,txtsnm,txtqty,txtcno,txtpric,txttot;

Choice br;

TextArea ta;

Button add,edit,search,update,delete,save,cancel;

int r = 170, g = 201, b = 204;

Color c;

mysupplier()

{

setTitle("Insert Supplier");

c = new Color(r,g,b);

setBackground(c);

l1 = new Label("Supplier ID :");

l1.setFont(new Font("arial",Font.BOLD,12));

l2 = new Label("Name :");

l2.setFont(new Font("arial",Font.BOLD,12));

l3 = new Label("Brand :");

l3.setFont(new Font("arial",Font.BOLD,12));

l4 = new Label("Address :");

l4.setFont(new Font("arial",Font.BOLD,12));

l5 = new Label("Quantity :");

l5.setFont(new Font("arial",Font.BOLD,12));

l6 = new Label("Contact No :");

l6.setFont(new Font("arial",Font.BOLD,12));

//

l7 = new Label("Price :");

l7.setFont(new Font("arial",Font.BOLD,12));

l8 = new Label("Total :");

l8.setFont(new Font("arial",Font.BOLD,12));

txtsid = new TextField(5);

txtsid.setFont(new Font("arial",Font.BOLD,12));

txtsnm = new TextField(30);

txtsnm.setFont(new Font("arial",Font.BOLD,12));

txtqty = new TextField(15);

txtqty.setFont(new Font("arial",Font.BOLD,12));

txtcno = new TextField(15);

txtcno.setFont(new Font("arial",Font.BOLD,12));

//

txtpric = new TextField(15);

txtpric.setFont(new Font("arial",Font.BOLD,12));

txttot = new TextField(10);

txttot.setFont(new Font("arial",Font.BOLD,12));

ta = new TextArea(2,30);

ta.setFont(new Font("arial",Font.BOLD,12));

br = new Choice();

br.setFont(new Font("arial",Font.BOLD,12));

add = new Button ("Add New");

add.setFont(new Font("arial",Font.BOLD,12));

edit = new Button ("Edit");

edit.setFont(new Font("arial",Font.BOLD,12));

save = new Button("Save");

save.setFont(new Font("arial",Font.BOLD,12));

update = new Button ("Update");

update.setFont(new Font("arial",Font.BOLD,12));

search = new Button("Search");

search.setFont(new Font("arial",Font.BOLD,12));

delete = new Button("Delete");

delete.setFont(new Font("arial",Font.BOLD,12));

cancel = new Button("Cancel");

cancel.setFont(new Font("arial",Font.BOLD,12));

add(l1);

add(l2);

add(l3);

add(l4);

add(l5);

add(l6);

//

add(l7);

add(txtpric);

add(l8);

add(txttot);

br.add(" ");

add(br);

add(txtsid);

add(txtsnm);

add(txtqty);

add(txtcno);

add(ta);

add(add);

add(edit);

add(search);

add(update);

add(save);

add(delete);

add(cancel);

br.addItemListener(this);

add.addActionListener(this);

edit.addActionListener(this);

update.addActionListener(this);

search.addActionListener(this);

delete.addActionListener(this);

save.addActionListener(this);

cancel.addActionListener(this);

// addWindowListener(new MyWindowAdapter());;

// addMouseListener(new MyMouseAdapter(this));

setLayout(new FlowLayout());

setSize(545,410);

txtsid.requestFocus();

br.setEnabled(false);

txtsnm.setEditable(true);

txtsnm.setEditable(false);

txtqty.setEditable(false);

//

txtpric.setEditable(false);

txttot.setEditable(false);

ta.setEditable(false);

txtcno.setEditable(false);

save.setEnabled(false);

delete.setEnabled(false);

update.setEnabled(false);

edit.setEnabled(false);

txtpric.addKeyListener (new KeyAdapter () {

public void keyTyped (KeyEvent ke)

{

char c = ke.getKeyChar ();

if (! ((Character.isDigit (c)) || (c == KeyEvent.VK\_BACK\_SPACE)))

{

getToolkit().beep ();

ke.consume ();

}

}

public void keyReleased(KeyEvent arg0)

{

int price=Integer.parseInt(txtpric.getText());//float

int qty=Integer.parseInt(txtqty.getText());//float

int tot=price\*qty;

txttot.setText(""+tot);

};

}

);

txtsnm.addKeyListener (new KeyAdapter () {

public void keyTyped (KeyEvent ke) {

char c = ke.getKeyChar ();

if (! ((Character.isLetter (c)) || (c == KeyEvent.VK\_BACK\_SPACE)||(c == KeyEvent.VK\_SPACE))) {

getToolkit().beep ();

ke.consume ();

//JOptionPane.showMessageDialog (null, "Enter only Alphabets");

}

}

}

);

txtsid.addKeyListener (new KeyAdapter () {

public void keyTyped (KeyEvent ke) {

char c = ke.getKeyChar ();

if (! ((Character.isDigit (c)) || (c == KeyEvent.VK\_BACK\_SPACE))) {

getToolkit().beep ();

ke.consume ();

}

}

}

);

txtqty.addKeyListener (new KeyAdapter () {

public void keyTyped (KeyEvent ke) {

char c = ke.getKeyChar ();

if (! ((Character.isDigit (c)) || (c == KeyEvent.VK\_BACK\_SPACE))) {

getToolkit().beep ();

ke.consume ();

}

}

}

);

txtcno.addKeyListener (new KeyAdapter ()

{

public void keyTyped (KeyEvent ke)

{

char c = ke.getKeyChar ();

if (! ((Character.isDigit (c)) || (c == KeyEvent.VK\_BACK\_SPACE)))

{

getToolkit().beep ();

ke.consume ();

}

}

}

);

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement stmt = conn.createStatement();

String cmdbr = "Select Distinct brand from product";

ResultSet rsbr = stmt.executeQuery(cmdbr);

while(rsbr.next())

{

br.addItem(rsbr.getString(1));

}//while

stmt.close();

conn.close();

}//try

catch(Exception e)

{

System.out.println(e);

}//catch

//////////////Supplier id choice box////

try {

Class.forName("com.mysql.jdbc.Driver");

} catch(java.lang.ClassNotFoundException e) {

System.err.print("ClassNotFoundException: ");

System.err.println(e.getMessage());

}

try {

con = DriverManager.getConnection(url,"root","");

stmt = con.createStatement();

} catch(SQLException ex) {

System.err.println("SQLException: " + ex.getMessage());

}

//Windoe hide//

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

public void paint(Graphics g)

{

g.drawString(mousemsg, mousex, mousey);

g.setColor(Color.black);

g.drawRoundRect(27,55,490,330,20,20);

l1.setLocation(40,70);

l2.setLocation(40,110);

l3.setLocation(250,70);

l4.setLocation(40,150);

l5.setLocation(40,220);

//

l7.setLocation(40,250);

txtpric.setLocation(125,250);

l6.setLocation(280,220);

txtsid.setLocation(135,70);

//

l8.setLocation(280,260);

txttot.setLocation(340,260);

txtsnm.setLocation(125,110);

br.setLocation(327,70);

ta.setLocation(125,150);

txtqty.setLocation(125,220);

txtcno.setLocation(375,220);

add.setLocation(130,290);

search.setLocation(205,290);

update.setLocation(280,290);

delete.setLocation(355,290);

save.setLocation(205,340);

cancel.setLocation(280,340);

edit.setLocation(130,340);

add.setSize(60,20);

search.setSize(60,20);

edit.setSize(60,20);

update.setSize(60,20);

delete.setSize(60,20);

save.setSize(60,20);

cancel.setSize(60,20);

}

public void itemStateChanged(ItemEvent ie)

{

repaint();

}

public void actionPerformed (ActionEvent ae)

{

String supid,sname,qty,cno,add,bran,pric,tot;

String str=ae.getActionCommand();

if(str.equals("Cancel"))

{

setVisible(false);

}

else if(str.equals("Add New"))

{

save.setEnabled(true);

Statement stmt1;

ResultSet rs1;

try

{

Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e)

{

System.out.println("Class not found"+e);

}

try

{

Connection conn1 = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

stmt1 =conn1.createStatement();

int temp=0;

rs1=stmt1.executeQuery("select \* from supplier where supp\_id=(select max(supp\_id) from supplier)");

txtsid.setText("");

txtsnm.setText("");

ta.setText("");

txtcno.setText("");

txtqty.setText("");

br.select(0);

txtsid.setEditable(true);

txtsnm.setEditable(true);

ta.setEditable(true);

txtcno.setEditable(true);

txtqty.setEditable(true);

txtpric.setEditable(true);

txttot.setEditable(true);

br.setEnabled(true);

save.setEnabled(true);

while(rs1.next())

{

temp=((rs1.getInt("supp\_id")) +1);

String t=Integer.toString(temp);

txtsid.setText(t);

}

}

catch(Exception e)

{

System.out.println("Driver not found"+e);

}

}

else if (str.equals("Save"))

{

txtsid.requestFocus ();

if (txtsid.getText().equals ("")||txtsnm.getText().equals ("")||txtqty.getText().equals ("")||br.getSelectedItem().equals ("")||txtcno.getText().equals ("")||ta.getText().equals ("") ||txtpric.getText().equals ("") ||txttot.getText().equals (""))

{

JOptionPane.showMessageDialog (this, "All Fields Required");

}

else{

supid=txtsid.getText();

sname=txtsnm.getText();

qty=txtqty.getText();

cno=txtcno.getText();

pric=txtpric.getText();

tot=txttot.getText();

add=ta.getText();

bran=br.getSelectedItem();

//

createString = "insert into supplier values('"+supid+"','"+sname+"','"+bran+"','"+add+"','"+qty+"','"+cno+"','"+pric+"','"+tot+"')";

JOptionPane.showMessageDialog(this,"Record saved Successfully");

txtClear ();

save.setEnabled(false);

try

{

stmt.executeUpdate(createString);

}

catch(SQLException ex)

{

System.err.println("SQLException "+ex.getMessage());

}

}

}

else if (str.equals("Edit"))

{

txtsnm.requestFocus();

txtsid.setEditable(false);

txtsnm.setEditable(true);

ta.setEditable(true);

txtcno.setEditable(true);

txtqty.setEditable(true);

txtpric.setEditable(true);

txttot.setEditable(true);

br.setEnabled(true);

update.setEnabled(true);

}

else

if(str.equals("Delete"))

{

createString = "delete from supplier where supp\_id = '"+txtsid.getText()+"'";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record Deleted Successfully");

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not Deleted ");

}

txtClear ();

delete.setEnabled(false);

edit.setEnabled(false);

}

else if(str.equals("Search"))

{

txtsid.requestFocus ();

if (txtsid.getText().equals (""))

{

JOptionPane.showMessageDialog (this, "Supplier ID Required");

}

else{

createString = "select \* from supplier where supp\_id like '"+txtsid.getText()+"'";

try

{

ResultSet rs = stmt.executeQuery(createString);

if(rs.next())

{

txtsid.setText(""+rs.getString(1));

txtsnm.setText(""+rs.getString(2));

br.select(""+rs.getString(3));

ta.setText(""+rs.getString(4));

txtqty.setText(""+rs.getInt(5));

txtcno.setText(""+rs.getInt(6));

txtpric.setText(""+rs.getInt(7));

txttot.setText(""+rs.getInt(8));

JOptionPane.showMessageDialog(this,"Record Searched Successfully");

}

else

JOptionPane.showMessageDialog(this,"Record not found");

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

}

//txtClear ();

delete.setEnabled(true);

edit.setEnabled(true);

save.setEnabled(false);

}

}

else if(str.equals(""))

{

createString = "update supplier set supp\_name = '"+txtsnm.getText()+"',brand='"+br.getSelectedItem()+"',add1 ='"+ta.getText()+"',quantity = '"+txtqty.getText()+"',contact\_no = '"+txtcno.getText()+"',price = '"+txtpric.getText()+"',Total = '"+txttot.getText()+"' where supp\_id like'"+txtsid.getText()+"'";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record Modified Successfully");

}

catch(SQLException ex)

{

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

}

txtClear ();

update.setEnabled(false);

edit.setEnabled(false);

}

repaint();

}

private void txtClear () {

txtsid.setText("");

txtsnm.setText("");

ta.setText("");

txtcno.setText("");

txtqty.setText("");

txtpric.setText("");

txttot.setText("");

br.select(0);

txtsid.setEditable(true);

br.setEnabled(false);

txtsnm.setEditable(false);

txtcno.setEditable(false);

txtqty.setEditable(false);

txtpric.setEditable(false);

txttot.setEditable(false);

ta.setEditable(false);

txtsid.requestFocus ();

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* View Suppiler \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class viewsupp extends Frame //implements ActionListener

{

TextArea ta;

Label l1,l2;

TextField txtsuppid;

Button Viewall,home;

int r=197,g=223,b=228;

Color c;

viewsupp()

{

c = new Color(r,g,b);

setBackground(c);

setSize(1000,700);

ta=new TextArea();

add(ta);

l1=new Label("All Supplier Details");

ta.setEditable(false);

add(l1);

////////////////////////

Viewall = new Button ("Viewall");

Viewall.setFont(new Font("arial",Font.BOLD,12));

home = new Button ("home");

home.setFont(new Font("arial",Font.BOLD,12));

add(Viewall);

add(home);

l2 = new Label("customer id ");

l2.setFont(new Font("arial",Font.BOLD,12));

txtsuppid = new TextField(15);

txtsuppid.setFont(new Font("arial",Font.BOLD,12));

add(l2);

add(txtsuppid);

l1.setFont(new Font("arial",Font.BOLD,25));

setTitle("View Supplier Details");

setLayout(new FlowLayout());

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement st=con.createStatement();

PreparedStatement ps;

ResultSet rs=null;

ps=con.prepareStatement("select \* from supplier ");

rs=ps.executeQuery();

setFont(new Font("arial",Font.BOLD,14));

String test = " \n \n\n\tCustomer ID\t\t\tName\t\t\tBrand\t\tAddress\t\tFax No\tContact No\n\n\n ";

ta.setText(test);

while(rs.next())

{

String SID=rs.getString(1);

String SNM=rs.getString(2);

String BRAN = rs.getString(3);

String ADD = rs.getString(4);

int FAX = rs.getInt(5);

int CNO = rs.getInt(6);

setFont(new Font("arial",Font.BOLD,12));

test = test+"\n\n\t "+ SID+"\t\t\t"+SNM+"\t\t\t"+BRAN+"\t\t"+ADD+"\t\t "+FAX+"\t "+CNO+"\n ";

}

ta.setText(test);

}

catch(Exception e)

{

System.out.println(e);

}

// addMouseListener(new MyMouseAdapter(this));

//Windoe hide//

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

public void paint(Graphics g)

{

// g.drawString(mousemsg, mousex, mousey);

ta.setLocation(100,200);

ta.setSize(850,450);

l1.setLocation(400,150);

/////////////////

l2.setLocation(100, 100);

txtsuppid.setLocation(150,100);

Viewall.setLocation(500,100);

home.setLocation(700,100);

Viewall.setSize(60,20);

home.setSize(60,20);

}

/\* public static void main(String args[])

{

viewsupp my=new viewsupp();

//my.setSize(800,700);

my.setVisible(true);

my.setTitle("View Supplier");

}

\*/

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Insert Customer \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class mycustomer extends Frame implements ActionListener

{

String url = "jdbc:mysql://localhost/WineS";

Connection con;

String createString;

Statement stmt;

String mousemsg=" ";

int mousex = 0, mousey = 0;

Label l1,l2,l3,l4,l5;

TextField txtcid,txtcnm,txtcno,txtdate;

TextArea ta;

Button add,update,delete,search,save,cancel,edit;

int r = 170, g = 201, b = 204;

Color c;

mycustomer()

{

setTitle("Insert Customer");

c = new Color(r,g,b);

setBackground(c);

l1 = new Label("Customer ID :");

l1.setFont(new Font("arial",Font.BOLD,12));

////////

l5 = new Label("Date ");

l5.setFont(new Font("arial",Font.BOLD,12));

txtdate = new TextField(5);

txtdate.setFont(new Font("arial",Font.BOLD,12));

l2 = new Label("Customer Nmae :");

l2.setFont(new Font("arial",Font.BOLD,12));

l3 = new Label("Address :");

l3.setFont(new Font("arial",Font.BOLD,12));

l4 = new Label("Contact No :");

l4.setFont(new Font("arial",Font.BOLD,12));

txtcid = new TextField(5);

txtcid.setFont(new Font("arial",Font.BOLD,12));

txtcnm = new TextField(30);

txtcnm.setFont(new Font("arial",Font.BOLD,12));

txtcno = new TextField(20);

txtcno.setFont(new Font("arial",Font.BOLD,12));

ta = new TextArea(2,30);

ta.setFont(new Font("arial",Font.BOLD,12));

add = new Button ("Add New");

add.setFont(new Font("arial",Font.BOLD,12));

edit = new Button ("Edit");

edit.setFont(new Font("arial",Font.BOLD,12));

save = new Button("Save");

save.setFont(new Font("arial",Font.BOLD,12));

update = new Button ("Update");

update.setFont(new Font("arial",Font.BOLD,12));

search = new Button("Search");

search.setFont(new Font("arial",Font.BOLD,12));

delete = new Button("Delete");

delete.setFont(new Font("arial",Font.BOLD,12));

cancel = new Button("Cancel");

cancel.setFont(new Font("arial",Font.BOLD,12));

add(l1);

add(l2);

add(l3);

add(l4);

add(l5);

add(txtdate);

add(txtcid);

add(txtcnm);

add(txtcno);

add(ta);

add(add);

add(edit);

add(search);

add(update);

add(save);

add(delete);

add(cancel);

add.addActionListener(this);

update.addActionListener(this);

edit.addActionListener(this);

search.addActionListener(this);

delete.addActionListener(this);

save.addActionListener(this);

cancel.addActionListener(this);

// addWindowListener(new MyWindowAdapter());;

// addMouseListener(new MyMouseAdapter(this));

setLayout(new FlowLayout());

setSize(495,410);

txtcid.requestFocus();

txtcid.setEditable(true);

txtcnm.setEditable(false);

ta.setEditable(false);

txtcno.setEditable(false);

save.setEnabled(false);

delete.setEnabled(false);

update.setEnabled(false);

edit.setEnabled(false);

txtcnm.addKeyListener (new KeyAdapter () {

public void keyTyped (KeyEvent ke) {

char c = ke.getKeyChar ();

if (! ((Character.isLetter (c)) || (c == KeyEvent.VK\_BACK\_SPACE)||(c == KeyEvent.VK\_SPACE))) {

getToolkit().beep ();

ke.consume ();

//JOptionPane.showMessageDialog (null, "Enter only Alphabets");

}

}

}

);

txtcid.addKeyListener (new KeyAdapter () {

public void keyTyped (KeyEvent ke) {

char c = ke.getKeyChar ();

if (! ((Character.isDigit (c)) || (c == KeyEvent.VK\_BACK\_SPACE))) {

getToolkit().beep ();

ke.consume ();

}

}

}

);

txtcno.addKeyListener (new KeyAdapter () {

public void keyTyped (KeyEvent ke) {

char c = ke.getKeyChar ();

if (! ((Character.isDigit (c)) || (c == KeyEvent.VK\_BACK\_SPACE))) {

getToolkit().beep ();

ke.consume ();

}

}

}

);

try {

Class.forName("com.mysql.jdbc.Driver");

} catch(java.lang.ClassNotFoundException e) {

System.err.print("ClassNotFoundException: ");

System.err.println(e.getMessage());

}

try {

con = DriverManager.getConnection(url,"root","");

stmt = con.createStatement();

} catch(SQLException ex) {

System.err.println("SQLException: " + ex.getMessage());

}

//Windoe hide//

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

public void paint(Graphics g)

{

g.drawString(mousemsg, mousex, mousey);

g.setColor(Color.black);

g.drawRoundRect(27,55,435,330,20,20);

l1.setLocation(50,70);

l5.setLocation(300,70);

txtdate.setLocation(500,70);

l2.setLocation(50,110);

l3.setLocation(50,150);

l4.setLocation(50,220);

txtcid.setLocation(190,70);

txtcnm.setLocation(190,110);

ta.setLocation(190,150);

txtcno.setLocation(190,220);

add.setLocation(105,290);

search.setLocation(180,290);

update.setLocation(255,290);

delete.setLocation(330,290);

save.setLocation(180,331);

cancel.setLocation(255,331);

edit.setLocation(105,331);

add.setSize(60,20);

edit.setSize(60,20);

search.setSize(60,20);

update.setSize(60,20);

delete.setSize(60,20);

save.setSize(60,20);

cancel.setSize(60,20);

}

public void actionPerformed (ActionEvent ae)

{

String cid,cname,cno,add;

String str=ae.getActionCommand();

if(str.equals("Cancel"))

{

setVisible(false);

}

else if(str.equals("Add New"))

{

save.setEnabled(true);

Statement stmt1;

ResultSet rs1;

try

{

Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e)

{

System.out.println("Class not found"+e);

}

try

{

Connection conn1 = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

stmt1 =conn1.createStatement();

int temp=0;

rs1=stmt1.executeQuery("select \* from customer where cust\_id=(select max(cust\_id) from customer)");

txtcid.setText(" ");

txtcnm.setText(" ");

ta.setText(" ");

txtcno.setText(" ");

txtcid.setEditable(true);

txtcnm.setEditable(true);

ta.setEditable(true);

txtcno.setEditable(true);

save.setEnabled(true);

while(rs1.next())

{

temp=((rs1.getInt("cust\_id")) +1);

String t=Integer.toString(temp);

txtcid.setText(t);

}

}

catch(Exception e)

{

System.out.println("Driver not found"+e);

}

}

else if (str.equals("Save"))

{

cid=txtcid.getText();

cname=txtcnm.getText();

cno=txtcno.getText();

add=ta.getText();

createString = "insert into customer values('"+cid+"','"+cname+"','"+add+"','"+cno+"')";

txtClear ();

save.setEnabled(false);

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record saved Successfully");

}

catch(SQLException ex)

{

System.err.println("SQLException "+ex.getMessage());

}

}

else if (str.equals("Edit"))

{

txtcnm.requestFocus();

txtcid.setEditable(false);

txtcnm.setEditable(true);

ta.setEditable(true);

txtcno.setEditable(true);

update.setEnabled(true);

}

else

if(str.equals("Delete"))

{

createString = "delete from customer where cust\_id = '"+txtcid.getText()+"'";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record Deleted Successfully");

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not Deleted ");

}

//txtClear ();

delete.setEnabled(false);

edit.setEnabled(false);

}

else if(str.equals("Search"))

{

txtcid.requestFocus ();

if (txtcid.getText().equals (""))

{

JOptionPane.showMessageDialog (this, "Customer ID Required");

}

else{

createString = "select \* from customer where cust\_id like'"+txtcid.getText()+"'";

try

{

ResultSet rs = stmt.executeQuery(createString);

if(rs.next())

{

txtcid.setText(""+rs.getInt(1));

txtcnm.setText(""+rs.getString(2));

ta.setText(""+rs.getString(3));

txtcno.setText(""+rs.getInt(4));

JOptionPane.showMessageDialog(this,"Record Searched Successfully");

}

else

JOptionPane.showMessageDialog(this,"Record not found ");

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not found ");

}

//txtClear ();

delete.setEnabled(true);

edit.setEnabled(true);

save.setEnabled(false);

}

}

else if(str.equals("Update"))

{

int CID = Integer.parseInt (txtcid.getText ());

createString = "update customer set cust\_name = '"+txtcnm.getText()+"',address ='"+ta.getText()+"',contact\_no = '"+txtcno.getText()+"' where cust\_id = '"+CID+"'";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record Modified Successfully");

}

catch(SQLException ex)

{

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

JOptionPane.showMessageDialog(this,"Record not Modified ");

}

txtClear ();

update.setEnabled(false);

edit.setEnabled(false);

}

repaint();

}

private void txtClear () {

txtcid.setText("");

txtcnm.setText("");

txtcno.setText("");

ta.setText("");

txtcid.requestFocus ();

txtcid.setEditable(true);

txtcnm.setEditable(false);

txtcno.setEditable(false);

ta.setEditable(false);

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* View Customer \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class viewcust extends Frame //implements ActionListener

{

TextArea ta;

Label l1,l2,l3;

TextField txtcid,txtcname;

Button Viewall,home;

int r=197,g=223,b=228;

Color c;

viewcust()

{

c = new Color(r,g,b);

setBackground(c);

setSize(1000,700);

ta=new TextArea();

add(ta);

l1=new Label("All Customer Details");

///////

Viewall = new Button ("Viewall");

Viewall.setFont(new Font("arial",Font.BOLD,12));

home = new Button ("home");

home.setFont(new Font("arial",Font.BOLD,12));

add(Viewall);

add(home);

l2 = new Label("customer id ");

l2.setFont(new Font("arial",Font.BOLD,12));

l3 = new Label("customer name ");

l3.setFont(new Font("arial",Font.BOLD,12));

txtcid = new TextField(10);

txtcid.setFont(new Font("arial",Font.BOLD,12));

txtcname = new TextField(15);

txtcname.setFont(new Font("arial",Font.BOLD,12));

add(l2);

add(txtcid);

add(l3);

add(txtcname);

ta.setEditable(false);

add(l1);

l1.setFont(new Font("arial",Font.BOLD,25));

setTitle("View Customer Details");

setLayout(new FlowLayout());

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement st=con.createStatement();

PreparedStatement ps;

ResultSet rs=null;

ps=con.prepareStatement("select \* from customer ");

rs=ps.executeQuery();

setFont(new Font("arial",Font.BOLD,14));

String test = " \n\n\n\tCustomer ID\t\t\t Name\t\t\t\t Address\t\t\t Contact No\n\n\n ";

ta.setText(test);

while(rs.next())

{

String CID=rs.getString(1);

String CNM=rs.getString(2);

String ADD = rs.getString(3);

int CNO = rs.getInt(4);

setFont(new Font("arial",Font.BOLD,12));

test = test+"\n\n\t "+ CID+"\t\t\t"+CNM+"\t\t\t\t"+ADD+"\t\t\t\t"+CNO+"\t\t\t\n ";

}

ta.setText(test);

}

catch(Exception e)

{

System.out.println(e);

}

// addMouseListener(new MyMouseAdapter(this));

//Windoe hide//

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

public void paint(Graphics g)

{

// g.drawString(mousemsg, mousex, mousey);

ta.setLocation(100,200);

ta.setSize(850,450);

l1.setLocation(400,150);

/////////

l2.setLocation(100, 100);

txtcid.setLocation(200,100);

l3.setLocation(300, 100);

txtcname.setLocation(420,100);

Viewall.setLocation(600,100);

home.setLocation(800,100);

Viewall.setSize(60,20);

home.setSize(60,20);

}

/\*public static void main(String args[])

{

viewcust my=new viewcust();

//my.setSize(800,700);

my.setVisible(true);

my.setTitle("View Customer");

}

\*/

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Purchase Order \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class mypurchase extends Frame implements ActionListener,ItemListener

{

String url = "jdbc:mysql://localhost/WineS";

Connection con;

String createString;

Statement stmt;

String mousemsg=" ";

int mousex = 0, mousey = 0;

Label l1,l2,l3,l4,l5,l6,l7,l8;

TextField txtpoid,txtdate,txtpnm,txtqnt,txtsnm;

Choice pid,siz,sid;

Button add,save,update,search,delete,cancel,edit;

int r = 170, g = 201, b = 204;

Color c;

//////

mypurchase()

{

setTitle("Purchase Order");

c = new Color(r,g,b);

setBackground(c);

l1 = new Label("Purchase Order ID :");

l1.setFont(new Font("arial",Font.BOLD,12));

l2 = new Label("Date :");

l2.setFont(new Font("arial",Font.BOLD,12));

l3 = new Label("Product Code");

l3.setFont(new Font("arial",Font.BOLD,12));

l4 = new Label("Product Name");

l4.setFont(new Font("arial",Font.BOLD,12));

l5 = new Label("Supplier ID :");

l5.setFont(new Font("arial",Font.BOLD,12));

l6 = new Label("Supplier Name");

l6.setFont(new Font("arial",Font.BOLD,12));

l7 = new Label("Size");

l7.setFont(new Font("arial",Font.BOLD,12));

l8 = new Label("Quantity");

l8.setFont(new Font("arial",Font.BOLD,12));

txtpoid = new TextField(3);

txtpoid.setFont(new Font("arial",Font.BOLD,12));

txtsnm = new TextField(27);

txtsnm.setFont(new Font("arial",Font.BOLD,12));

txtdate = new TextField(7);

txtdate.setFont(new Font("arial",Font.BOLD,12));

txtpnm = new TextField(27);

txtpnm.setFont(new Font("arial",Font.BOLD,12));

txtqnt = new TextField(5);

txtqnt.setFont(new Font("arial",Font.BOLD,12));

////DAte///////

DateTimeFormatter dtf = DateTimeFormatter.ofPattern("yyyy/MM/dd");

LocalDate localDate = LocalDate.now();

System.out.println(dtf.format(localDate));

txtdate.setText(localDate.toString());

siz = new Choice();

siz.setFont(new Font("arial",Font.BOLD,12));

pid=new Choice ();

pid.setFont(new Font("arial",Font.BOLD,12));

sid=new Choice ();

sid.setFont(new Font("arial",Font.BOLD,12));

add = new Button ("ADD");

add.setFont(new Font("arial",Font.BOLD,12));

edit = new Button ("Edit");

edit.setFont(new Font("arial",Font.BOLD,12));

save = new Button("Save");

save.setFont(new Font("arial",Font.BOLD,12));

update = new Button("Update");

update.setFont(new Font("arial",Font.BOLD,12));

search = new Button("Search");

search.setFont(new Font("arial",Font.BOLD,12));

delete = new Button("Delete");

delete.setFont(new Font("arial",Font.BOLD,12));

cancel = new Button("Cancel");

cancel.setFont(new Font("arial",Font.BOLD,12));

add(l1);

add(l2);

add(l3);

add(l4);

add(l5);

add(l6);

add(l7);

add(l8);

siz.add(" ");

add(siz);

////adding choice control////////

siz.addItem("60ml");

siz.addItem("90ml");

siz.addItem("180ml");

siz.addItem("275ml");

siz.addItem("330ml");

siz.addItem("375ml");

siz.addItem("500ml");

siz.addItem("750ml");

siz.addItem("1000ml");

siz.addItem("2000ml");

pid.addItem(" ");

add(pid);

sid.addItem(" ");

add(sid);

add(txtpoid);

add(txtdate);

add(txtpnm);

add(txtqnt);

add(txtsnm);

add(add);

add(edit);

add(save);

add(update);

add(search);

add(delete);

add(cancel);

siz.addItemListener(this);

sid.addItemListener(this);

pid.addItemListener(this);

add.addActionListener(this);

edit.addActionListener(this);

save.addActionListener(this);

update.addActionListener(this);

search.addActionListener(this);

delete.addActionListener(this);

cancel.addActionListener(this);

// addWindowListener(new MyWindowAdapter());;

// addMouseListener(new MyMouseAdapter(this));

setLayout(new FlowLayout());

setSize(480,550);

txtpoid.requestFocus();

txtdate.setEditable(false);

siz.setEnabled(false);

pid.setEnabled(false);

sid.setEnabled(false);

txtqnt.setEditable(false);

txtsnm.setEditable(false);

txtpnm.setEditable(false);

save.setEnabled(false);

delete.setEnabled(false);

update.setEnabled(false);

edit.setEnabled(false);

//////Supplier choice box//////////

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement stmt = conn.createStatement();

String cmdsup = "Select Distinct supp\_id from supplier";

ResultSet rssup = stmt.executeQuery(cmdsup);

while(rssup.next())

{

sid.addItem(rssup.getString(1));

}//while

stmt.close();

conn.close();

}//try

catch(Exception e)

{

System.out.println(e);

}//catch

//////Supplier choice box//////////

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement stmt = conn.createStatement();

String cmdsup = "Select Distinct pro\_code from product";

ResultSet rssup = stmt.executeQuery(cmdsup);

while(rssup.next())

{

pid.addItem(rssup.getString(1));

}//while

stmt.close();

conn.close();

}//try

catch(Exception e)

{

System.out.println(e);

}//catch

try

{

Class.forName("com.mysql.jdbc.Driver");

}

catch(java.lang.ClassNotFoundException e)

{

System.err.print("ClassNotFoundException: ");

System.err.println(e.getMessage());

}

try

{

con = DriverManager.getConnection(url,"root","");

stmt = con.createStatement();

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

}

//Windoe hide//

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

public void paint(Graphics g)

{

g.drawString(mousemsg, mousex, mousey);

g.setColor(Color.black);

g.drawRoundRect(27,55,420,360,20,20);

l1.setLocation(65,80);

l2.setLocation(65,120);

l3.setLocation(65,240);

l4.setLocation(65,280);

l5.setLocation(65,160);

l6.setLocation(65,200);

l7.setLocation(65,320);

l8.setLocation(65,360);

txtpoid.setLocation(220,80);

txtdate.setLocation(203,120);

pid.setLocation(203,240);

txtpnm.setLocation(203,280);

sid.setLocation(203,160);

txtsnm.setLocation(203,200);

siz.setLocation(203,320);

txtqnt.setLocation(203,360);

add.setSize(60,20);

edit.setSize(60,20);

search.setSize(60,20);

update.setSize(60,20);

delete.setSize(60,20);

save.setSize(60,20);

cancel.setSize(60,20);

add.setLocation(90,450);

edit.setLocation(250,450);

save.setLocation(170,500);

update.setLocation(330,450);

search.setLocation(170,450);

delete.setLocation(90,500);

cancel.setLocation(250,500);

}

public void itemStateChanged(ItemEvent ie)

{ String PName,SName;

if(ie.getSource()==pid)

{

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

String cmdNameDetails = "Select \* from product where pro\_code = ?";

PreparedStatement stmtNameDetails = conn.prepareStatement(cmdNameDetails);

stmtNameDetails.setString(1,(String)pid.getSelectedItem());

ResultSet rsProName = stmtNameDetails.executeQuery();

while(rsProName.next())

{

PName = rsProName.getString(2);

txtpnm.setText(PName);

siz.select(rsProName.getString(5));

//System.out.println(rsProName.getString(5));

txtqnt.setText(rsProName.getString(7));

}

}

catch(Exception e)

{

System.out.println(e);

}

}

if(ie.getSource()==sid)

{

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

String cmdNameDetails = "Select supp\_name from supplier where supp\_id = ?";

PreparedStatement stmtNameDetails = conn.prepareStatement(cmdNameDetails);

stmtNameDetails.setString(1,(String)sid.getSelectedItem());

ResultSet rsSupName = stmtNameDetails.executeQuery();

while(rsSupName.next())

{

SName = rsSupName.getString(1);

txtsnm.setText(SName);

}

}

catch(Exception e)

{

System.out.println(e);

}

}

repaint();

}

public void actionPerformed (ActionEvent ae)

{

String SName,PName,POID,PDATE,PID,SID,SIZ,QNT;

POID=txtpoid.getText();

PDATE=txtdate.getText();

PID=(String)pid.getSelectedItem();

SID=(String)sid.getSelectedItem();

SIZ=siz.getSelectedItem();

QNT=txtqnt.getText();

String str = ae.getActionCommand();

if(str.equals("ADD"))

{

save.setEnabled(true);

Statement stmt1;

ResultSet rs1;

try

{

Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e)

{

System.out.println("Class not found"+e);

}

try

{

Connection conn1 = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

stmt1 =conn1.createStatement();

int temp=0;

rs1=stmt1.executeQuery("select \* from purchase\_order where pur\_id=(select max(pur\_id) from purchase\_order)");

txtpoid.setText("");

txtdate.setText("");

sid.select(0);

pid.select(0);

siz.select(0);

txtqnt.setText("");

txtsnm.setText("");

txtpnm.setText("");

save.setEnabled(true);

txtpoid.requestFocus();

txtdate.setEditable(true);

siz.setEnabled(true);

pid.setEnabled(true);

sid.setEnabled(true);

txtqnt.setEditable(true);

txtsnm.setEditable(true);

txtpnm.setEditable(true);

save.setEnabled(true);

while(rs1.next())

{

temp=((rs1.getInt("pur\_id")) +1);

String t=Integer.toString(temp);

txtpoid.setText(t);

}

}

catch(Exception e)

{

System.out.println("Driver not found"+e);

}

}

else if (str.equals("Edit"))

{

txtdate.requestFocus();

txtpoid.setEditable(false);

pid.setEnabled(true);

sid.setEnabled(true);

siz.setEnabled(true);

txtpnm.setEditable(true);

txtsnm.setEditable(true);

txtdate.setEditable(true);

txtqnt.setEditable(true);

update.setEnabled(true);

}

else if (str.equals("Save"))

{

txtpoid.requestFocus ();

if (txtpoid.getText().equals ("")||txtdate.getText().equals ("")||pid.getSelectedItem().equals ("")||sid.getSelectedItem().equals ("")||siz.getSelectedItem().equals ("")||txtqnt.getText().equals ("")||txtsnm.getText().equals ("")||txtpnm.getText().equals (""))

{

JOptionPane.showMessageDialog (this, "All Fields Required");

}

else{

createString = "insert into purchase\_order values('"+POID+"','"+PDATE+"','"+PID+"','"+SID+"','"+SIZ+"','"+QNT+"')";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record saved Successfully");

}

catch(SQLException ex)

{

System.err.println("SQLException "+ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not saved ");

}

txtpoid.setText("");

txtdate.setText("");

sid.select(0);

pid.select(0);

siz.select(0);

txtqnt.setText("");

txtsnm.setText("");

txtpnm.setText("");

txtpoid.setEditable(true);

txtdate.setEditable(false);

pid.setEnabled(false);

sid.setEnabled(false);

siz.setEnabled(false);

txtqnt.setEditable(false);

txtpoid.requestFocus();

save.setEnabled(false);

}

}

else if(str.equals("Search"))

{

txtpoid.requestFocus ();

if (txtpoid.getText().equals (""))

{

JOptionPane.showMessageDialog (this, "Purchse O. Code Required");

}

else{

try

{

createString = "select \* from purchase\_order where pur\_id like '"+txtpoid.getText()+"'";

ResultSet rs = stmt.executeQuery(createString);

if(rs.next())

{

txtpoid.setText(""+rs.getString(1));

txtdate.setText(""+rs.getString(2));

pid.select(""+rs.getString(3));

sid.select(""+rs.getString(4));

siz.select(""+rs.getString(5));

txtqnt.setText(""+rs.getInt(6));

JOptionPane.showMessageDialog(this,"Record Searched Successfully");

}

else

JOptionPane.showMessageDialog(this,"Record not found");

txtpoid.setEditable(true);

txtdate.setEditable(false);

pid.setEnabled(false);

sid.setEnabled(false);

siz.setEnabled(false);

txtqnt.setEditable(false);

delete.setEnabled(true);

edit.setEnabled(true);

save.setEnabled(false);

txtpoid.requestFocus();

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not found");

}

}

}

else if(str.equals("Update"))

{

createString = "update purchase\_order set pur\_date = '"+PDATE+"',pur\_code ='"+PID+"',supp\_id ='"+SID+"',size ='"+SIZ+"',quantity = '"+QNT+"' where pur\_id = '"+POID+"'";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record Modified Successfully");

}

catch(SQLException ex)

{

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

JOptionPane.showMessageDialog(this,"Record not Modified ");

}

txtpoid.setText("");

txtdate.setText("");

sid.select(0);

pid.select(0);

siz.select(0);

txtqnt.setText("");

txtsnm.setText("");

txtpnm.setText("");

txtpoid.setEditable(true);

txtdate.setEditable(false);

pid.setEnabled(false);

sid.setEnabled(false);

siz.setEnabled(false);

txtqnt.setEditable(false);

txtpoid.requestFocus();

update.setEnabled(false);

edit.setEnabled(false);

}

else

if(str.equals("Delete"))

{

createString = "delete from purchase\_order where pur\_id = '"+POID+"'";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record Deleted Successfully");

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not Deleted ");

}

txtpoid.setText("");

txtdate.setText("");

sid.select(0);

pid.select(0);

siz.select(0);

txtqnt.setText("");

txtsnm.setText("");

txtpnm.setText("");

txtpoid.setEditable(true);

txtdate.setEditable(false);

pid.setEnabled(false);

sid.setEnabled(false);

siz.setEnabled(false);

txtqnt.setEditable(false);

txtpoid.requestFocus();

delete.setEnabled(false);

edit.setEnabled(false);

}

if(str.equals("Cancel"))

{

setVisible(false);

}

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* View Purchase Order \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class viewpo extends Frame //implements ActionListener

{

String mousemsg = " ";

int mousex = 0, mousey = 0;

Label l1;

int r=197,g=223,b=228;

Color c;

TextArea ta;

viewpo()

{

c = new Color(r,g,b);

setBackground(c);

setSize(1000,700);

ta=new TextArea();

add(ta);

l1=new Label("All Purchase Order Details");

ta.setEditable(false);

add(l1);

l1.setFont(new Font("arial",Font.BOLD,25));

setTitle("View Purchase Order Details");

setLayout(new FlowLayout());

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement st=con.createStatement();

PreparedStatement ps;

ResultSet rs=null;

ps=con.prepareStatement("select \* from purchase\_order ");

rs=ps.executeQuery();

setFont(new Font("arial",Font.BOLD,14));

String test = " \n\n\n\tPO ID\t\tDate\t Supp.ID\t Pro.ID\t\t Size\t\t\tQuantity\n\n\n";

ta.setText(test);

while(rs.next())

{

String POID = rs.getString(1);

String PDATE=rs.getString(2);

String PID = rs.getString(3);

String SID = rs.getString(4);

String SIZ=rs.getString(5);

int QNT = rs.getInt(6);

setFont(new Font("arial",Font.BOLD,12));

test = test+"\n\t"+ POID+"\t\t"+PDATE+"\t "+SID+ "\t\t "+PID+ "\t\t"+SIZ+ "\t\t\t "+QNT+"\n ";

}

ta.setText(test);

}

catch(Exception e)

{

System.out.println(e);

}

// addMouseListener(new MyMouseAdapter(this));

//Windoe hide//

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

public void paint(Graphics g)

{

g.drawString(mousemsg, mousex, mousey);

ta.setLocation(100,200);

ta.setSize(850,450);

l1.setLocation(400,150);

}

/\*public static void main(String args[])

{

viewpo my=new viewpo();

//my.setSize(800,700);

my.setVisible(true);

my.setTitle("View Purchase Order Information");

}

\*/

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Receipt \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class myreceipt extends Frame implements ActionListener,ItemListener

{

String url = "jdbc:mysql://localhost/WineS";

Connection con;

String createString;

Statement stmt;

String mousemsg=" ";

int mousex = 0, mousey = 0;

Label l1,l2,l3,l4,l5,l6,l7,l8,l9,l10,l11,l12;

TextField txtrid,txtdate,txtcnm,txtpnm,txtqnt,txtpric,txttotal;

Choice siz,pid,cid;

Button total,add,save,update,search,delete,cancel,edit;

int r=197,g=223,b=228;

Color c;

myreceipt()

{

setTitle("Receipt");

c = new Color(r,g,b);

setBackground(c);

l1 = new Label("Receipt No");

l1.setFont(new Font("arial",Font.BOLD,12));

l2 = new Label("Date :");

l2.setFont(new Font("arial",Font.BOLD,12));

l3 = new Label("Customer ID");

l3.setFont(new Font("arial",Font.BOLD,12));

l4 = new Label("Nmae");

l4.setFont(new Font("arial",Font.BOLD,12));

l5 = new Label("Product Code :");

l5.setFont(new Font("arial",Font.BOLD,12));

l6 = new Label("Name");

l6.setFont(new Font("arial",Font.BOLD,12));

l7 = new Label("Size");

l7.setFont(new Font("arial",Font.BOLD,12));

l8 = new Label("Quantity");

l8.setFont(new Font("arial",Font.BOLD,12));

l9 = new Label("Price");

l9.setFont(new Font("arial",Font.BOLD,12));

l10 = new Label("Receipt Details");

l10.setFont(new Font("arial",Font.BOLD,18));

l11 = new Label("Customer Details");

l11.setFont(new Font("arial",Font.BOLD,12));

l12 = new Label("Order Details");

l12.setFont(new Font("arial",Font.BOLD,12));

txtrid = new TextField(3);

txtrid.setFont(new Font("arial",Font.BOLD,12));

txtcnm = new TextField(27);

txtcnm.setFont(new Font("arial",Font.BOLD,12));

txtdate = new TextField(7);

txtdate.setFont(new Font("arial",Font.BOLD,12));

txtpnm = new TextField(27);

txtpnm.setFont(new Font("arial",Font.BOLD,12));

txtqnt = new TextField(5);

txtqnt.setFont(new Font("arial",Font.BOLD,12));

txtpric = new TextField(5);

txtpric.setFont(new Font("arial",Font.BOLD,12));

txttotal = new TextField(5);

txttotal.setFont(new Font("arial",Font.BOLD,12));

DateTimeFormatter dtf = DateTimeFormatter.ofPattern("yyyy/MM/dd");

LocalDate localDate = LocalDate.now();

System.out.println(dtf.format(localDate));

txtdate.setText(localDate.toString());

siz = new Choice();

siz.setFont(new Font("arial",Font.BOLD,12));

pid=new Choice ();

pid.setFont(new Font("arial",Font.BOLD,12));

cid=new Choice ();

cid.setFont(new Font("arial",Font.BOLD,12));

total = new Button ("Total Amount");

total.setFont(new Font("arial",Font.BOLD,12));

add = new Button ("ADD");

add.setFont(new Font("arial",Font.BOLD,12));

edit = new Button ("Edit");

edit.setFont(new Font("arial",Font.BOLD,12));

save = new Button("Save");

save.setFont(new Font("arial",Font.BOLD,12));

update = new Button("Update");

update.setFont(new Font("arial",Font.BOLD,12));

search = new Button("Search");

search.setFont(new Font("arial",Font.BOLD,12));

delete = new Button("Delete");

delete.setFont(new Font("arial",Font.BOLD,12));

cancel = new Button("Cancel");

cancel.setFont(new Font("arial",Font.BOLD,12));

add(l1);

add(l2);

add(l3);

add(l4);

add(l5);

add(l6);

add(l7);

add(l8);

add(l9);

add(l10);

add(l11);

add(l12);

siz.add(" ");

add(siz);

////adding choice control////////

siz.addItem("60ml");

siz.addItem("90ml");

siz.addItem("180ml");

siz.addItem("275ml");

siz.addItem("330ml");

siz.addItem("375ml");

siz.addItem("500ml");

siz.addItem("750ml");

siz.addItem("1000ml");

siz.addItem("2000ml");

pid.addItem("");// changes kelay

add(pid);

cid.addItem("");

add(cid);

add(txtrid);

add(txtdate);

add(txtcnm);

add(txtpnm);

add(txtqnt);

add(txtpric);

add(txttotal);

add(total);

add(add);

add(edit);

add(save);

add(update);

add(search);

add(delete);

add(cancel);

siz.addItemListener(this);

cid.addItemListener(this);

pid.addItemListener(this);

total.addActionListener(this);

add.addActionListener(this);

edit.addActionListener(this);

save.addActionListener(this);

update.addActionListener(this);

search.addActionListener(this);

delete.addActionListener(this);

cancel.addActionListener(this);

addWindowListener(new MyWindowAdapter());;

// addMouseListener(new MyMouseAdapter(this));

setLayout(new FlowLayout());

setSize(670,470);

//////Customer choice box//////////

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement stmt = conn.createStatement();

String cmdsup = "Select Distinct cust\_id from customer";

ResultSet rssup = stmt.executeQuery(cmdsup);

while(rssup.next())

{

cid.addItem(rssup.getString(1));

}//while

stmt.close();

conn.close();

}//try

catch(Exception e)

{

System.out.println(e);

}//catch

//////Product choice box//////////

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement stmt = conn.createStatement();

String cmdsup = "Select Distinct pro\_code from product";

ResultSet rssup = stmt.executeQuery(cmdsup);

while(rssup.next())

{

pid.addItem(rssup.getString(1));

}//while

stmt.close();

conn.close();

}//try

catch(Exception e)

{

System.out.println(e);

}//catch

////////////////////////////

try

{

Class.forName("com.mysql.jdbc.Driver");

}

catch(java.lang.ClassNotFoundException e)

{

System.err.print("ClassNotFoundException: ");

System.err.println(e.getMessage());

}

try

{

con = DriverManager.getConnection(url,"root","");

stmt = con.createStatement();

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

}

}

public void paint(Graphics g)

{

g.drawString(mousemsg, mousex, mousey);

g.setColor(Color.black);

//g.drawRoundRect(27,55,420,360,20,20);

g.drawRoundRect(27,60,610,380,0,0);

g.drawRoundRect(47,130,560,60,0,0);

g.drawRoundRect(47,215,560,160,0,0);

l1.setLocation(68,80);

l2.setLocation(300,80);

l3.setLocation(68,148);

l4.setLocation(300,148);

l5.setLocation(68,235);

l6.setLocation(300,235);

l7.setLocation(68,275);

l8.setLocation(300,275);

l9.setLocation(68,315);

l10.setLocation(40,45);

l11.setLocation(68,118);

l12.setLocation(68,204);

txtrid.setLocation(180,80);

txtdate.setLocation(370,80);

cid.setLocation(180,148);

txtcnm.setLocation(370,148);

pid.setLocation(180,235);

txtpnm.setLocation(370,235);

siz.setLocation(180,275);

txtqnt.setLocation(370,275);

txtpric.setLocation(180,315);

txttotal.setLocation(400,315);

add.setSize(60,25);

edit.setSize(60,25);

search.setSize(60,25);

update.setSize(60,25);

delete.setSize(60,25);

save.setSize(60,25);

cancel.setSize(60,25);

total.setLocation(300,315);

add.setLocation(60,400);

edit.setLocation(220,400);

save.setLocation(460,400);

update.setLocation(300,400);

search.setLocation(140,400);

delete.setLocation(380,400);

cancel.setLocation(540,400);

}

public void itemStateChanged(ItemEvent ie)

{

String CName,PName;

if(ie.getSource()==pid)

{

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

String cmdNameDetails = "Select \* from product where pro\_code = ?";

PreparedStatement stmtNameDetails = conn.prepareStatement(cmdNameDetails);

stmtNameDetails.setString(1,(String)pid.getSelectedItem());

ResultSet rsProName = stmtNameDetails.executeQuery();

while(rsProName.next())

{

PName = rsProName.getString(1);

txtpnm.setText(PName);

siz.select(rsProName.getString(5));

//System.out.println(rsProName.getString(5));

txtqnt.setText(rsProName.getString(7));

txtpric.setText(rsProName.getString(6));

}

}

catch(Exception e)

{

System.out.println(e);

}

}

if(ie.getSource()==cid)

{

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection conn = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

String cmdNameDetails = "Select cust\_name from customer where cust\_id = ?";

PreparedStatement stmtNameDetails = conn.prepareStatement(cmdNameDetails);

stmtNameDetails.setString(1,(String)cid.getSelectedItem());

ResultSet rsCustName = stmtNameDetails.executeQuery();

while(rsCustName.next())

{

CName = rsCustName.getString(1);

txtcnm.setText(CName);

}

}

catch(Exception e)

{

System.out.println(e);

}

}

repaint();

}

public void actionPerformed (ActionEvent ae)

{

String CName,PName,RID,SDATE,CID,PID,SIZ,QNT,PRC,TOT;

RID=txtrid.getText();

SDATE=txtdate.getText();

CID=(String)cid.getSelectedItem();

PID=(String)pid.getSelectedItem();

SIZ=siz.getSelectedItem();

QNT=txtqnt.getText();

PRC=txtpric.getText();

TOT=txttotal.getText();

String str = ae.getActionCommand();

if(str.equals("ADD"))

{

save.setEnabled(true);

Statement stmt1;

ResultSet rs1;

try

{

Class.forName("com.mysql.jdbc.Driver");

}

catch(Exception e)

{

System.out.println("Class not found"+e);

}

try

{

Connection conn1 = DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

stmt1 =conn1.createStatement();

int temp=0;

rs1=stmt1.executeQuery("select \* from receipt where receipt\_no=(select max(receipt\_no) from receipt)");

txtcnm.setText("");

txtpnm.setText("");

txtrid.setText("");

txtdate.setText("");

txtqnt.setText("");

txtpric.setText("");

txttotal.setText("");

siz.select(0);

cid.select(0);

pid.select(0);

txtcnm.setEditable(true);

txtpnm.setEditable(true);

txtrid.setEditable(true);

txtdate.setEditable(true);

txtqnt.setEditable(true);

txtpric.setEditable(true);

txttotal.setEditable(true);

siz.setEnabled(true);

cid.setEnabled(true);

pid.setEnabled(true);

save.setEnabled(true);

while(rs1.next())

{

temp=((rs1.getInt("receipt\_no")) +1);

String t=Integer.toString(temp);

txtrid.setText(t);

}

}

catch(Exception e)

{

System.out.println("Driver not found"+e);

}

}

else if (str.equals("Edit"))

{

txtdate.requestFocus();

txtrid.setEditable(false);

pid.setEnabled(true);

cid.setEnabled(true);

siz.setEnabled(true);

txtpnm.setEditable(true);

txtcnm.setEditable(true);

txtdate.setEditable(true);

txtqnt.setEditable(true);

txtpric.setEditable(true);

txttotal.setEditable(true);

update.setEnabled(true);

}

else if (str.equals("Save"))

{

txtrid.requestFocus ();

if (RID.equals ("")||SDATE.equals ("")||CID.equals ("")||PID.equals ("")||SIZ.equals ("")||QNT.equals ("")||PRC.equals ("")||TOT.equals (""))

{

JOptionPane.showMessageDialog (this, "All Fields Required");

}

else{

createString = "insert into receipt values('"+RID+"','"+SDATE+"','"+CID+"','"+PID+"','"+SIZ+"','"+QNT+"','"+PRC+"','"+TOT+"')";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record saved Successfully");

}

catch(SQLException ex)

{

System.err.println("SQLException "+ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not saved ");

}

txtrid.setText("");

txtdate.setText("");

cid.select(0);

pid.select(0);

siz.select(0);

txtqnt.setText("");

txtcnm.setText("");

txtpnm.setText("");

txtpric.setText("");

txttotal.setText("");

txtrid.setEditable(true);

txtdate.setEditable(false);

cid.setEnabled(false);

pid.setEnabled(false);

siz.setEnabled(false);

txtqnt.setEditable(false);

txtpric.setEditable(false);

txttotal.setEditable(false);

txtrid.requestFocus();

save.setEnabled(false);

}

}

else if(str.equals("Search"))

{

txtrid.requestFocus ();

if (txtrid.getText().equals (""))

{

JOptionPane.showMessageDialog (this, "Receipt No Required");

}

else{

try

{

createString = "select \* from receipt where receipt\_no like '"+txtrid.getText()+"'";

ResultSet rs = stmt.executeQuery(createString);

if(rs.next())

{

txtrid.setText(""+rs.getString(1));

txtdate.setText(""+rs.getString(2));

cid.select(""+rs.getString(3));

pid.select(""+rs.getString(4));

siz.select(""+rs.getString(5));

txtqnt.setText(""+rs.getInt(6));

txtpric.setText(""+rs.getInt(7));

txttotal.setText(""+rs.getInt(8));

JOptionPane.showMessageDialog(this,"Record Searched Successfully");

}

else

JOptionPane.showMessageDialog(this,"Record not found");

txtrid.setEditable(true);

txtdate.setEditable(false);

pid.setEnabled(false);

cid.setEnabled(false);

siz.setEnabled(false);

txtqnt.setEditable(false);

txtpric.setEditable(false);

txttotal.setEditable(false);

txtpnm.setEditable(false);

txtcnm.setEditable(false);

delete.setEnabled(true);

edit.setEnabled(true);

save.setEnabled(false);

txtrid.requestFocus();

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not found");

}

}

}

else if(str.equals("Update"))

{

createString = "update receipt set sal\_date = '"+SDATE+"',cust\_id ='"+CID+"',pro\_code ='"+PID+"',size ='"+SIZ+"',quantity = '"+QNT+"',price ='"+PRC+"',total = '"+TOT+"' where receipt\_no = '"+RID+"'";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record Modified Successfully");

}

catch(SQLException ex)

{

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

JOptionPane.showMessageDialog(this,"Record not Modified ");

}

txtrid.setText("");

txtdate.setText("");

cid.select(0);

pid.select(0);

siz.select(0);

txtqnt.setText("");

txtcnm.setText("");

txtpnm.setText("");

txtpric.setText("");

txttotal.setText("");

txtrid.setEditable(true);

txtdate.setEditable(false);

cid.setEnabled(false);

pid.setEnabled(false);

siz.setEnabled(false);

txtqnt.setEditable(false);

txtpric.setEditable(false);

txttotal.setEditable(false);

txtrid.requestFocus();

update.setEnabled(false);

edit.setEnabled(false);

}

else

if(str.equals("Delete"))

{

createString = "delete from receipt where receipt\_no = '"+RID+"'";

try

{

stmt.executeUpdate(createString);

JOptionPane.showMessageDialog(this,"Record Deleted Successfully");

}

catch(SQLException ex)

{

System.err.println("SQLException: " + ex.getMessage());

JOptionPane.showMessageDialog(this,"Record not Deleted ");

}

txtrid.setText("");

txtdate.setText("");

cid.select(0);

pid.select(0);

siz.select(0);

txtqnt.setText("");

txtcnm.setText("");

txtpnm.setText("");

txtpric.setText("");

txttotal.setText("");

txtrid.setEditable(true);

txtdate.setEditable(false);

cid.setEnabled(false);

pid.setEnabled(false);

siz.setEnabled(false);

txtqnt.setEditable(false);

txtpric.setEditable(false);

txttotal.setEditable(false);

txtrid.requestFocus();

delete.setEnabled(false);

edit.setEnabled(false);

}

else

if (str.equals("Total Amount"))

{

txttotal.setText("");

int total;

total = Integer.parseInt(txtqnt.getText()) \* Integer.parseInt(txtpric.getText());

String stotal = txttotal.getText();

stotal = stotal + total;

txttotal.setText(stotal);

}

if(str.equals("Cancel"))

{

setVisible(false);

}

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* View Sales \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class viewsales extends Frame //implements ActionListener

{

String mousemsg = " ";

int mousex = 0, mousey = 0;

Label l1,l2;

TextField txtdate;

Button Viewall,home;

int r=197,g=223,b=228;

Color c;

TextArea ta;

viewsales()

{

c = new Color(r,g,b);

setBackground(c);

setSize(1000,700);

ta=new TextArea();

add(ta);

l1=new Label("All Sales Details");

ta.setEditable(false);

add(l1);

/////////////////

Viewall = new Button ("Viewall");

Viewall.setFont(new Font("arial",Font.BOLD,12));

home = new Button ("home");

home.setFont(new Font("arial",Font.BOLD,12));

add(Viewall);

add(home);

l2 = new Label("customer id ");

l2.setFont(new Font("arial",Font.BOLD,12));

txtdate = new TextField(10);

txtdate.setFont(new Font("arial",Font.BOLD,12));

add(l2);

add(txtdate);

l1.setFont(new Font("arial",Font.BOLD,25));

setTitle("View Sales Details");

setLayout(new FlowLayout());

try

{

Class.forName("com.mysql.jdbc.Driver");

Connection con=DriverManager.getConnection("jdbc:mysql://localhost/WineS","root","");

Statement st=con.createStatement();

PreparedStatement ps;

ResultSet rs=null;

ps=con.prepareStatement("select \* from receipt ");

rs=ps.executeQuery();

setFont(new Font("arial",Font.BOLD,14));

String test = " \n\n\n\tRec. ID\t\tDate\t Cust ID\t\t\tProduct ID\t\t Size\tQuantity\t\tPrice\t Total\n\n\n ";

ta.setText(test);

while(rs.next())

{

String RID = rs.getString(1);

String RDATE=rs.getString(2);

String CID = rs.getString(3);

String PID = rs.getString(4);

String SIZ=rs.getString(5);

int QNT = rs.getInt(6);

int PRC = rs.getInt(7);

int TOT = rs.getInt(8);

setFont(new Font("arial",Font.BOLD,12));

test = test+"\n\t"+ RID+"\t\t" +RDATE+"\t "+CID+ "\t\t\t "+PID+"\t\t "+SIZ+ "\t "+QNT+ "\t\t"+PRC+ "\t "+TOT+"\n ";

}

ta.setText(test);

}

catch(Exception e)

{

System.out.println(e);

}

// addMouseListener(new MyMouseAdapter(this));

//Windoe hide//

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

public void paint(Graphics g)

{

g.drawString(mousemsg, mousex, mousey);

ta.setLocation(100,200);

ta.setSize(850,450);

l1.setLocation(400,150);

////////////

l2.setLocation(100, 100);

txtdate.setLocation(150,100);

Viewall.setLocation(500,100);

home.setLocation(700,100);

Viewall.setSize(60,20);

home.setSize(60,20);

}

/\* public static void main(String args[])

{

viewsales my=new viewsales();

//my.setSize(800,700);

my.setVisible(true);

my.setTitle("View Sales");

}

\*/

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Customer Reports \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class customerrep extends Frame //implements ActionListener

{

customerrep()

{

setSize(600,600);

setTitle("Customer Reports");

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Stock Report \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class stockrep extends Frame //implements ActionListener

{

stockrep()

{

setSize(600,600);

setTitle("Stock Reports");

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Supplier Report \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class supplierrep extends Frame //implements ActionListener

{

supplierrep()

{

setSize(600,600);

setTitle("Supplier Reports");

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Purchase Report \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class purchaserep extends Frame //implements ActionListener

{

purchaserep()

{

setSize(600,600);

setTitle("Purchase Reports");

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Sales Report \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class salesrep extends Frame //implements ActionListener

{

salesrep()

{

setSize(600,600);

setTitle("Sales Reports");

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

}

}

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* About \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

class help extends JFrame

{

JFrame f;

JLabel frontlabel;

help()

{

f=new JFrame("About");

setBounds(0,0,1024,768);

setDefaultCloseOperation(EXIT\_ON\_CLOSE);

ImageIcon ob1 = new ImageIcon("/root/Wineshop Management System1.jpg");

frontlabel = new JLabel(ob1);

f.add(frontlabel);

addWindowListener(new WindowAdapter()

{

public void windowClosing(WindowEvent e)

{

hide();

}

});

f.setResizable(false);

f.setVisible(true);

f.pack();

}

}