SHOPPING CART

DESCRIPTION

This is a project based on online shopping. In this new age, with more dreams and less time, people tend not to waste time waiting in lines to pay for their articles, thus they turn to online stores for their needs. This project is just a sample of the vast prospects in online shopping.

The user logs in or signs up to the store and is then taken to the categories page. The user can choose among various categories-clothes, watches and books. On clicking the image of the object the description of the item is displayed. Once an item is added to the cart, its quantity can be changed and it can be deleted. Attractive discounts are also offered on playing a game. Online shopping has the convenience of various modes of payment for example cash on delivery, credit/debit card.

This project comprises of 16 frames:

1. **JFrame**:
   - The frame to ask the user for login/sign-up
2. **Frame_login:**
   - Login for already existing user, by checking the corresponding password associated with the username in the 'shopping cart' table of the 'shopping' database in MySQL.
   - A table having the same name as the user name is created to store the details of selected items in cart. If a table with the same name already exists, that table is dropped.
3. **Frame_signup**:
   - Sign-up for a new user, the details of this user is stored in the 'shopping cart' table.
   - A table having the same name as the user name is created to store the details of selected items in cart. If a table with the same name already exists, that table is dropped.
4. **Frame_categories**:
   - Frame that displays available categories. User can select the items either through combo-box or the buttons displaying different images every second with the help of a timer.
5. **Frame_clothes_women**:

- Displays images of available items on buttons.
- Item details are displayed in a dialog box, if the user clicks on an item.
- An 'add to cart' button available for each item, if clicked a dialog box asks for the quantity of the corresponding item.
- These values are inserted into the table, named after the user in the 'shopping' database.

6. **Frame_clothes_men**:
    - Displays images of available items on buttons.
    - Item details are displayed in a dialog box, if the user clicks on an item.
    - An 'add to cart' button available for each item, if clicked a dialog box asks for the quantity of the corresponding item.
    - These values are inserted into the table, named after the user.
7. **Frame_clothes_kids**-
    - Displays images of available items on buttons.
    - Item details are displayed in a dialog box, if the user clicks on an item.
    - An 'add to cart' button available for each item, if clicked a dialog box asks for the quantity of the corresponding item.
    - These values are inserted into the table, named after the user.
8. **Frame_watch_men**-
    - Displays images of available items on buttons.
    - Item details are displayed in a dialog box, if the user clicks on an item.
    - An 'add to cart' button available for each item, if clicked a dialog box asks for the quantity of the corresponding item.
    - These values are inserted into the table, named after the user.
9. **Frame_watch_women**-
    - Displays images of available items on buttons.
    - Item details are displayed in a dialog box, if the user clicks on an item.
    - An 'add to cart' button available for each item, if clicked a dialog box asks for the quantity of the corresponding item.
    - These values are inserted into the table, named after the user.
10. **Frame_fictional**-
    - Displays images of available items on buttons.
    - Item details are displayed in a dialog box, if the user clicks on an item.
    - An 'add to cart' button available for each item, if clicked a dialog box asks for the quantity of the corresponding item.
    - These values are inserted into the table, named after the user.
11. **Frame_nonfic**-
    - Displays images of available items on buttons.
    - Item details are displayed in a dialog box, if the user clicks on an item.

- An ‘add to cart’ button available for each item, if clicked a dialog box asks for the quantity of the corresponding item.
- These values are inserted into the table, named after the user.

12. **Frame_cart**-
    - A default table model to display the contents of the cart, which may change dynamically.
    - An option to change the quantity of a selected item. The query is fired to MySQL and the table having the same name as username is updated.
    - An option to remove a selected item from the cart and this item is dropped from the table.
    - An option to play the ‘discount game’ if the total amount in cart exceeds Rs.8000. An option to pay the bill.
    - An option to go back to categories frame.
13. **Frame_discount** -
    - This frame comprises a game to get discount if the total amount in the cart exceeds Rs.8000.
    - The user is given 15 seconds to guess a maximum of 7 companies based on the logos displayed on the frame, for each correct answer the user gets 1% discount.
    - This counter of 15 seconds starts as soon as the frame is activated, the frame visibility is set false as soon as the counter runs down to 0.
14. **Frame_bill**-
    - The total amount and discounted amount are displayed.The user is asked for the mode of delivery.
15. **Frame_cod**-
    - The billing details are displayed. The expected date of delivery is also displayed, this is done by adding 5 days to the current date.
16. **Frame_credit**-
    - Card information is asked, this information is stored in a table ‘credit’.

Acknowledgement

In performing our assignment, we had to take the help and guideline of some respected persons, who deserve our greatest gratitude. The completion of this assignment gives us much pleasure. We would like to show our gratitude to our beloved teacher Mrs.Vibhavari for giving us such good guidance for this assignment throughout. We would also like to expand our deepest gratitude to our principal Mrs.C.V Madhvi for giving us this wonderful opportunity. In addition we would like to thank Mr.Prakash Kamble for his valuable suggestions.

Many people, especially our classmates and parents, have made valuable comments and suggestions on this project which gave us an inspiration to improve. We finally thank all the people for their help directly and indirectly to complete our assignment.

Bibliography

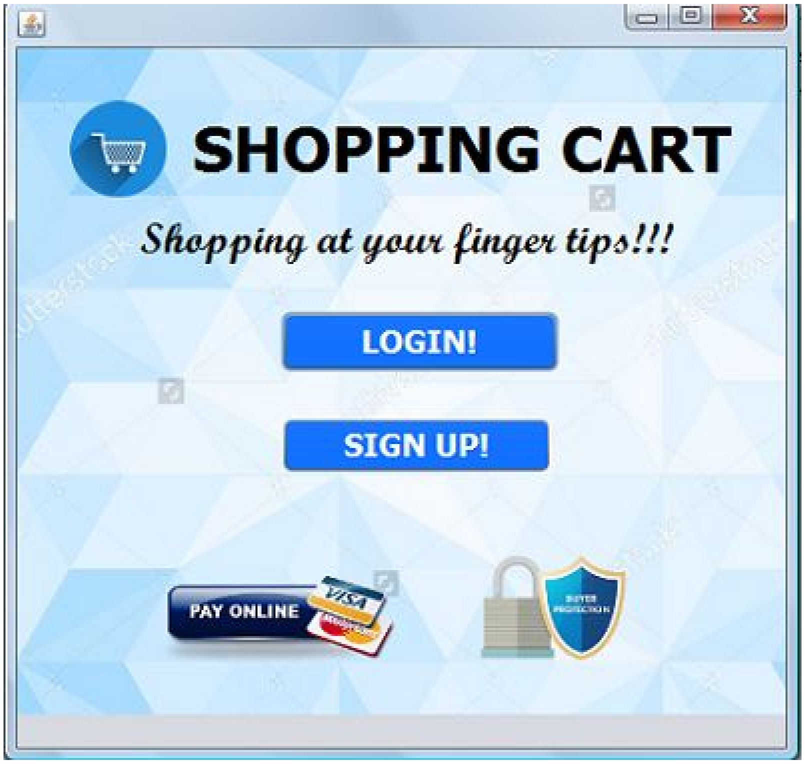
Sumita Aurora

www.w3school.com

Shopping Cart.

Imports:

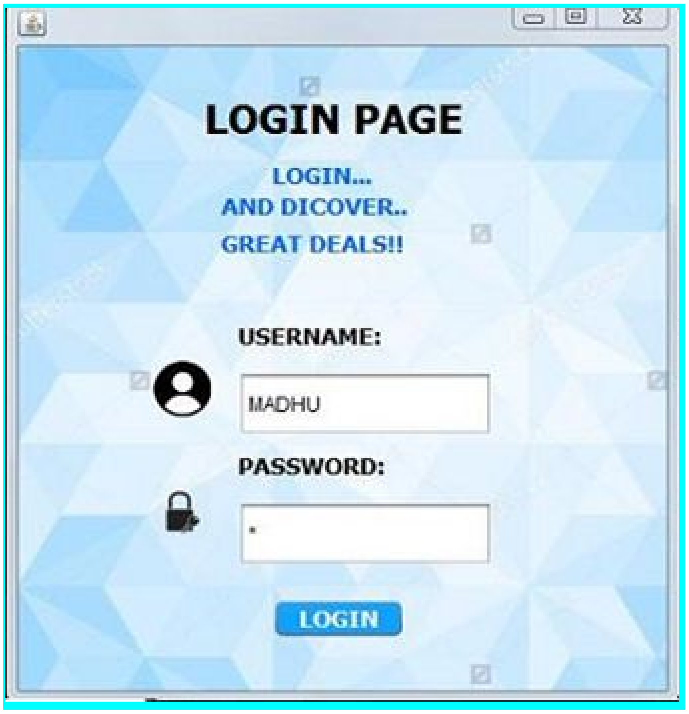
```
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.*;
import javax.swing.ImageIcon;
import javax.swing.JOptionPane;
import javax.swing.Timer;
import javax.swing.table.*;
```



```
public class frame_1 extends javax.swing.JFrame{
//global variables are intialised.
Connection con=null;
Statement st=null;
ResultSet rs=null;
int disc_pic=1;
int sec=30;
```

```
Double total_amt=0.0;
String cart_name="";
int a=0, b=0, c=0;
String disc_ans="";
Double disc=0.0;
double discounted_amt=0.0;
int total=0;
Double fin_amt=0.0;
int correct=0;
//Timers for displaying pictures in a loop.The pictures change every 2seconds.
Timer dod_t= new Timer(1000,new ActionListener(){
public void actionPerformed(ActionEvent evt){
a++;
//a is incremented by 1 every second.
if(a%2==0){button_dod.setIcon(new ImageIcon("C:\\IP PROJECT\\IP_1.png"));}
//Every time a is divisible by 2 img1 is displayed.As a is incremented every second the
images switch every 2 seconds.
else button_dod.setIcon(new ImageIcon("C:\\IP PROJECT\\IP_2.png"));}
// Every time a is not divisible by 2(ie is an odd no) img 2 is displayed.
});
Timer dod_t1= new Timer(1000,new ActionListener(){
public void actionPerformed(ActionEvent evt){
b++;
if(b%2==0){button_dod2.setIcon(new ImageIcon("C:\\IP PROJECT\\IP_3.jpg"));}
else button_dod2.setIcon(new ImageIcon("C:\\IP PROJECT\\IP_4.jpg"));
}});
Timer dod_t3= new Timer(1000,new ActionListener(){
public void actionPerformed(ActionEvent evt){
c++;
if(c%2==0){button_dod1.setIcon(new ImageIcon("C:\\IP PROJECT\\IP_5.jpg"));}
else button_dod1.setIcon(new ImageIcon("C:\\IP PROJECT\\IP_6.jpg"));
}});
//Timer used in the logo quiz like game for discount.It is a countdown timer of
20seconds.
Timer disc_game=new Timer(1000,new ActionListener(){
public void actionPerformed(ActionEvent e){
int sec=Integer.parseInt(disc_timer.getText());
sec--;
//1 is subtracted from sec every second.
disc_timer.setText(""+sec);
if(sec==0){disc_game.stop();
//once the countdown reaches 0,the timer stops and the bill frame is visible.
JOptionPane.showMessageDialog(null,"Discount earned="+disc+"%");
frame_bill.setVisible(true);
frame_discount.setVisible(false);
}}});
```

```
private void formWindowActivated(java.awt.event.WindowEvent evt) {
try{
Class.forName("java.sql.Driver");
con=DriverManager.getConnection("jdbc:mysql://localhost/shoppingcart","root","1234");
st=con.createStatement();
String sql=null;
//To establish jdbc connectivity.
}
catch(Exception e){JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void formWindowOpened(java.awt.event.WindowEvent evt) {
try{
Class.forName("java.sql.Driver");
con=DriverManager.getConnection("jdbc:mysql://localhost/mysql","root","1234");
st=con.createStatement();
String sql1="create database shoppingcart;";
String sql2="use shoppingcart;";
String sql3="create table shopping(name varchar(30),email varchar(50),username
varchar(30),password varchar(30));";
st.executeUpdate(sql1);
st.executeUpdate(sql2);
st.executeUpdate(sql3);
//Creates database “shoppingcart” and a table for usernames and passwords.
}
catch(Exception e){System.out.println("");
//Catch is empty because the database may already exist.
}}
private void login_btnActionPerformed(java.awt.event.ActionEvent evt){
frame_login.setVisible(true);
}
private void signup_btnActionPerformed(java.awt.event.ActionEvent evt) {
frame_signup.setVisible(true);
}
```



LOGIN PAGE

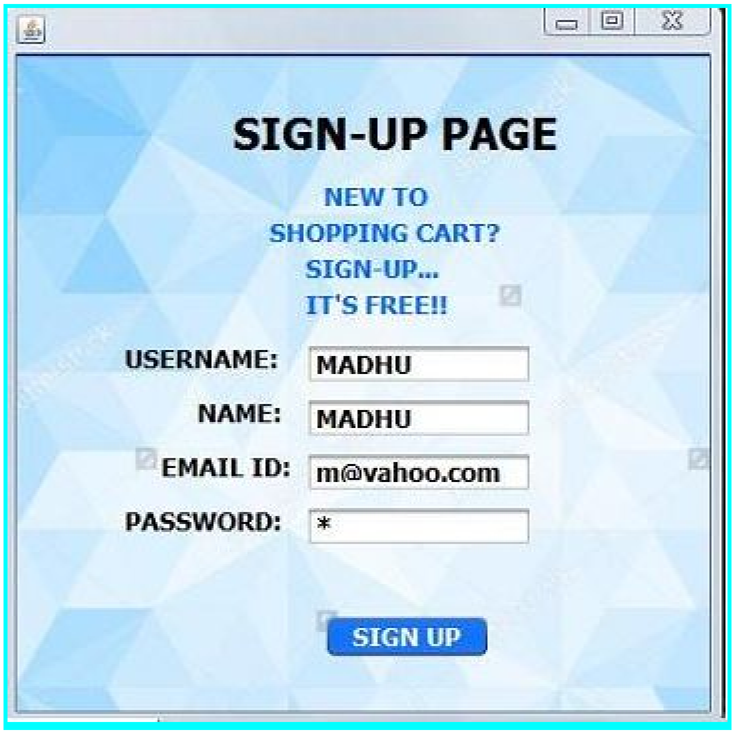
LOGIN...
AND DICOVER..
GREAT DEALS!!

USERNAME: MADHU

PASSWORD: *

Frame_login

```
private void loginActionPerformed(java.awt.event.ActionEvent evt) {
try{
String username=username_login.getText();
String password= new String(password_login.getPassword());
String sql1="use shoppingcart;";
int a= st.executeUpdate(sql1);
String sql=("select password from shopping where username='"+username+"' and
password='"+password+"';");
rs=st.executeQuery(sql);
if (rs.first()==false){
JOptionPane.showMessageDialog(null,"wrong username/password");}
else {
st.executeUpdate("drop table "+username+";");
//Incase the table already exists.
st.executeUpdate("create table "+username+" (item varchar(20),price int(10),qty
int(5),total float(10));");
frame_categories.setVisible(true);
frame_login.setVisible(false);
cart_name=username;
//a global variable,so can be used in other parts of the project.
}
//The categories frame opens only when the username and password match a row in
the result set.A table whose name is the username is created in the database.The items
“added to the cart” by the user will be stored in this table.
}
catch(Exception e){JOptionPane.showMessageDialog(null,e.getMessage());}
}
```



SIGN-UP PAGE

NEW TO
SHOPPING CART?
SIGN-UP...
IT'S FREE!!

USERNAME: MADHU

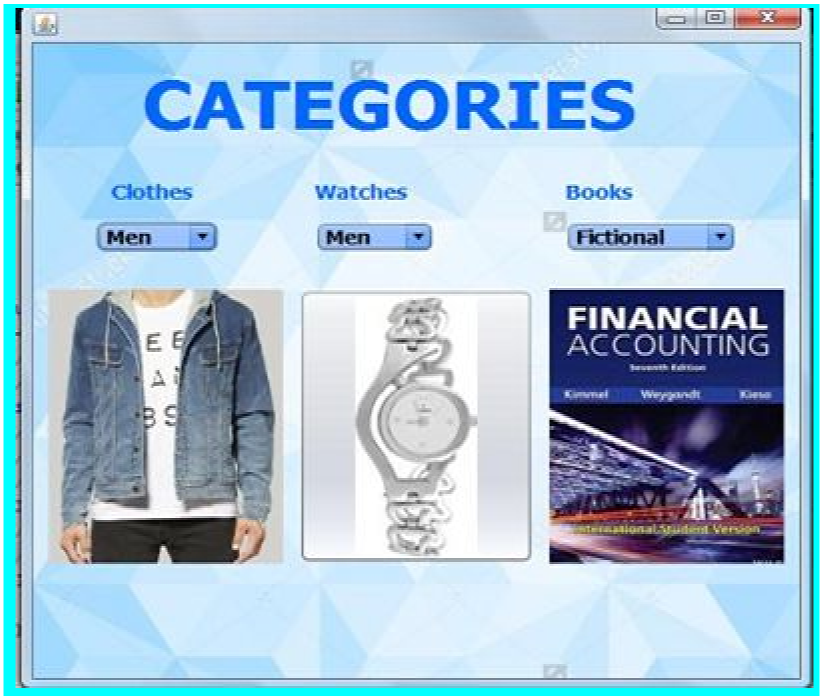
NAME: MADHU

EMAIL ID: m@vahoo.com

PASSWORD: *

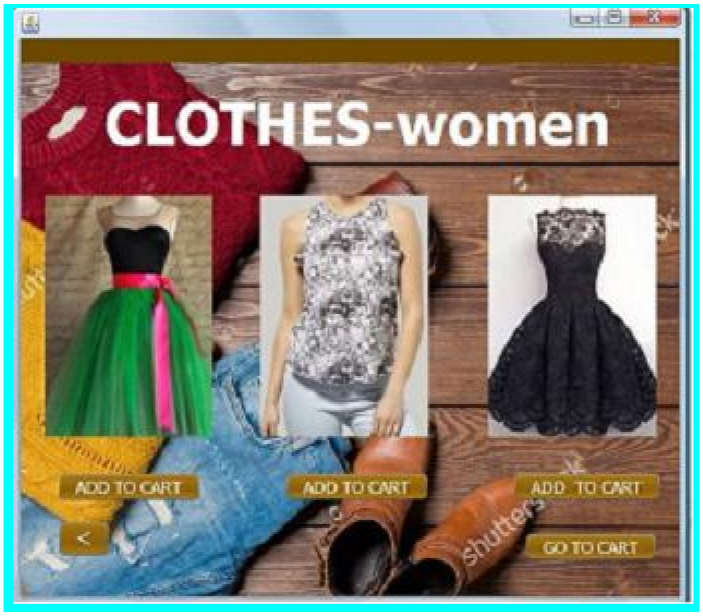
Frame_signup

```
private void sign_up1ActionPerformed(java.awt.event.ActionEvent evt){
try{
String name=name_signup.getText();
String email=eid_signup.getText();
String username=username_signup.getText();
String password=new String(password_signup.getPassword());
String sql=("insert into shopping
values('"+name+"','"+email+"','"+username+"','"+password+"');");
st.executeUpdate(sql);
JOptionPane.showMessageDialog(null,"account created.");
st.executeUpdate("create table "+username+" (item varchar(20),price int(10),qty
int(5),total float(10));");
cart_name=username;
frame_categories.setVisible(true);
frame_signup.setVisible(false);
//The user data is stored in a table shopping .And a table is created in the database with
the same name as the username.
}
catch(Exception e){JOptionPane.showMessageDialog(null,e.getMessage());}
}
```



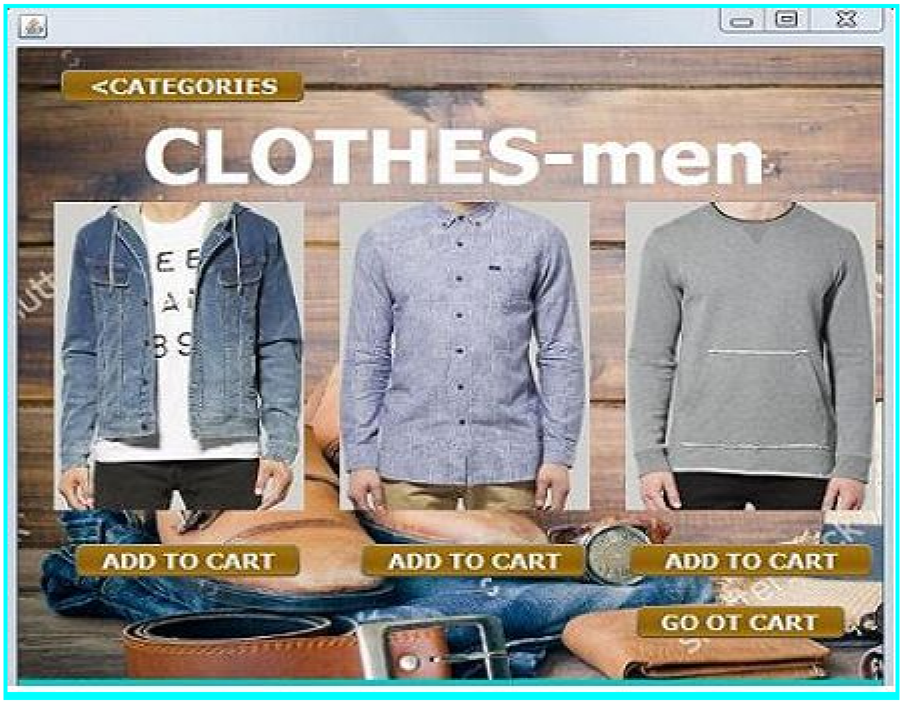
Frame_categories

```
private void frame_categoriesWindowActivated(java.awt.event.WindowEvent evt)
{
dod_t.start();
dod_t1.start();
dod_t3.start();
//Timers are activated for displaying two pictures in the buttons in a loop.
}
private void button_dod1ActionPerformed(java.awt.event.ActionEvent evt){
if(a%2==0){frame_clothes_women.setVisible(true);}
//Every time a is a even number(ie divisible by 2)the img on the button will correspond to women’s clothes.otherwise to men’s clothes.
else frame_clothes_men.setVisible(true);
frame_categories.setVisible(false);
}
private void button_dod2ActionPerformed(java.awt.event.ActionEvent evt){
if(b%2==0){frame_watch_women.setVisible(true);}
else frame_watch_men.setVisible(true);
frame_categories.setVisible(false);
}
private void button_dod3ActionPerformed(java.awt.event.ActionEvent evt) {
if(c%2==0){frame_fictional.setVisible(true);}
else frame_nonfic.setVisible(true);
frame_categories.setVisible(false);
}
private void cb_clothesActionPerformed(java.awt.event.ActionEvent evt) {
String sel_item=(cb_clothes.getSelectedItem()).toString();
if(sel_item.equals("Men")){frame_clothes_men.setVisible(true);}
if(sel_item.equals("Women")){frame_clothes_women.setVisible(true);}
if(sel_item.equals("Kids")){frame_clothes_kids.setVisible(true);}
frame_categories.setVisible(false);
}
private void cb_watchesActionPerformed(java.awt.event.ActionEvent evt) {
String sel_item=(cb_watches.getSelectedItem()).toString();
if(sel_item.equals("Men")){frame_watch_men.setVisible(true);}
if(sel_item.equals("Women")){frame_watch_women.setVisible(true);}
frame_categories.setVisible(false);
}


private void cb_booksActionPerformed(java.awt.event.ActionEvent evt) {
String sel_item=(cb_books.getSelectedItem()).toString();
if(sel_item.equals("Fictional"))frame_fictional.setVisible(true);
if(sel_item.equals("Non-Fiction"))frame_nonfic.setVisible(true);
frame_categories.setVisible(false);
}
```



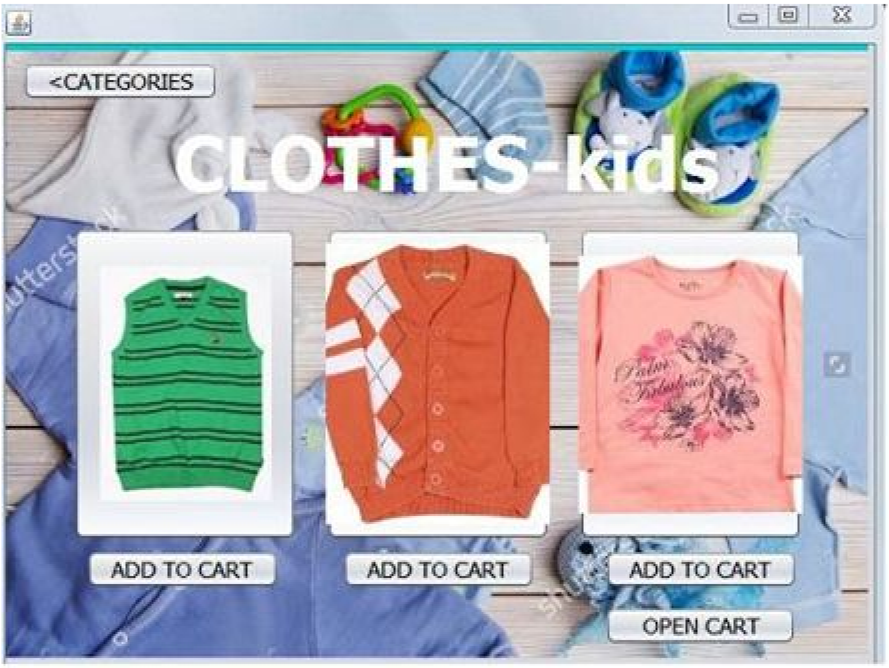
Frame_clothes_women

```
private void clothes_women_b1ActionPerformed(java.awt.event.ActionEvent evt)
{
JOptionPane.showMessageDialog(null,"Description:Grey Dress Size:M Price:2000");
//Displays the information about the product via a message dialog box.
}
private void clothes_women_b2ActionPerformed(java.awt.event.ActionEvent evt)
{
JOptionPane.showMessageDialog(null,"Description:Black Top Size:L Price:8000");
}
private void clothes_women_b3ActionPerformed(java.awt.event.ActionEvent evt)
{
JOptionPane.showMessageDialog(null,"Description:Black Dress Size:M Price:4000");
}
private void add_wcloth1ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"qty","Item Qty",3));
//The quantity of the item the user wants to buy is entered vis a InputDialog box.
int tot=qty*2000;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values
('GreenDress',2000,"+qty+","+tot+");");
//The item name,price,quantity and total.
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void add_wcloth2ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity","Item
Qty",3));
int tot=qty*900;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('black Top',900,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void add_wcloth3ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*4000;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values
('BlackDress',4000,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void back_wclothActionPerformed(java.awt.event.ActionEvent evt) {
frame_categories.setVisible(true);
frame_clothes_women.setVisible(false);
}

private void opencart_cloth_womenActionPerformed(java.awt.event.ActionEvent
evt) {
frame_cart.setVisible(true);
frame_clothes_women.setVisible(false);
}
```



Frame_clothes_men

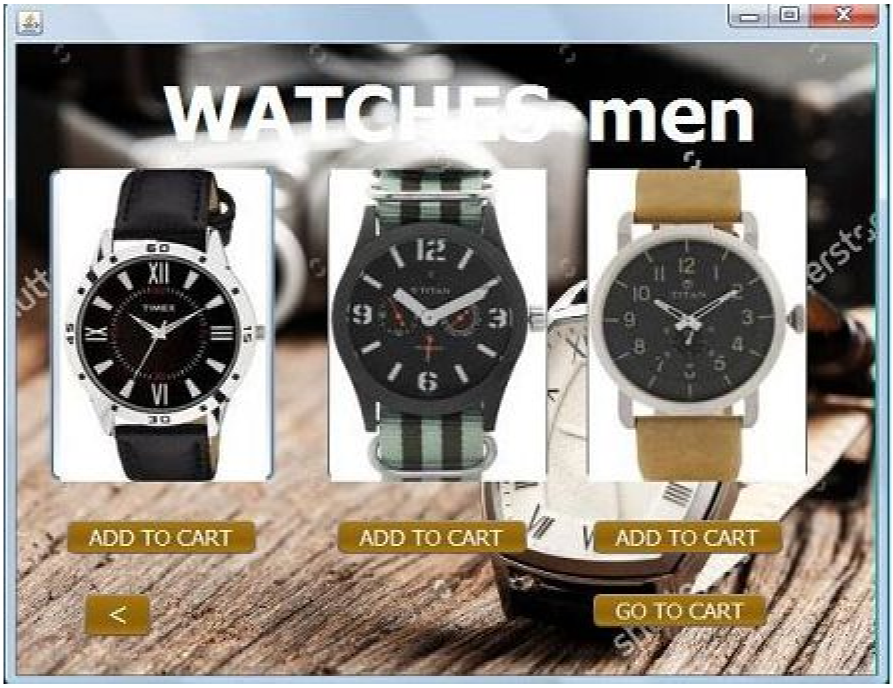
```
private void cloths_men_b1ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Jean Jacket Size:M Price:1200");
}
private void cloths_men_b2ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:blue shirt Size:L Price:900");
}
private void cloths_men_b3ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Grey Tshirt Size:M Price:1000");}
private void add_clothes_men1ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*1200;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Jean
Jacket',1200,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void add_clothes_men2ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*900;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('blue shirt',900,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void add_clothes_men3ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*1000;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Grey
Tshirt',1000,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void back_mclothActionPerformed(java.awt.event.ActionEvent evt) {
frame_categories.setVisible(true);
frame_clothes_men.setVisible(false);
}
private void opencart_cloth_menActionPerformed(java.awt.event.ActionEvent
evt) {
frame_cart.setVisible(true);
frame_clothes_men.setVisible(false);
}
```



Frame_clothes_kids

```
private void clothes_kid_b1ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Green Sweater Size:L Price:750");
}
private void clothes_kid_b2ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Orange sweater Size:L Price:800");
}
private void clothes_kid_b3ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Peach sweatshirt Size:M
Price:900");
}
private void add_clothes_kid1ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*750;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Green
Sweater',750,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
```

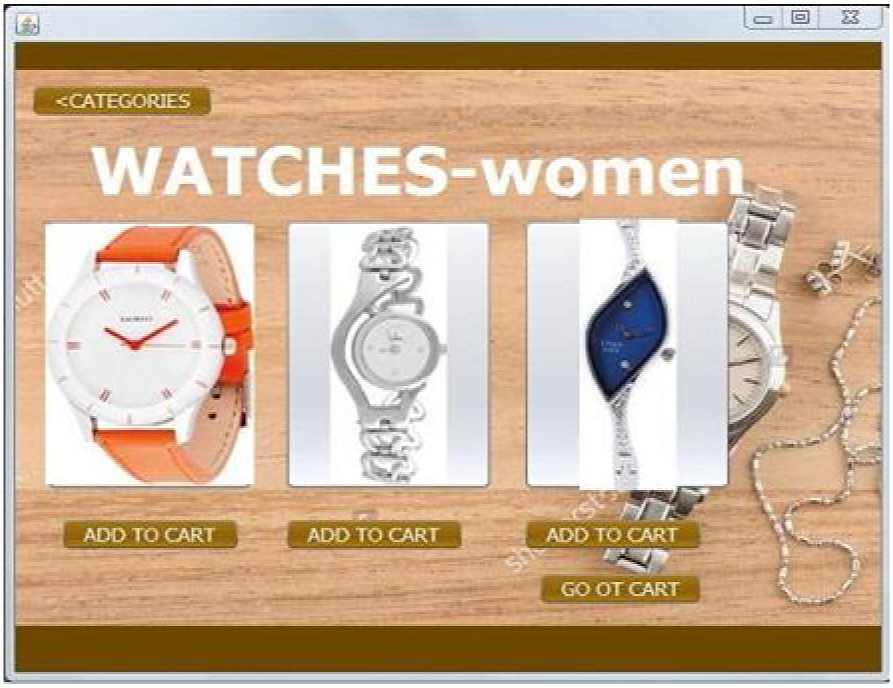
```
private void add_clothes_kid2ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*800;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Orange
Sweater',800,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}

private void add_clothes_kid3ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*900;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Peach
Sweatshirt',900,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void back_kclothActionPerformed(java.awt.event.ActionEvent evt) {
frame_categories.setVisible(true);
frame_clothes_kids.setVisible(false);
}
private void opencart_cloth_kidActionPerformed(java.awt.event.ActionEvent evt)
{
frame_cart.setVisible(true);
frame_clothes_kids.setVisible(false);}
```



Frame_watch_men

```
private void watch_men_b1ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Timex Black Price:1000");
}
private void watch_men_b2ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Titan Green Price:1500");
}
private void watch_men_b3ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Titan Brown Price:2000");
}
private void add_watch_men1ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*1000;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Timex
Black',1000,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void add_watch_men2ActionPerformed(java.awt.event.ActionEvent evt) {
```

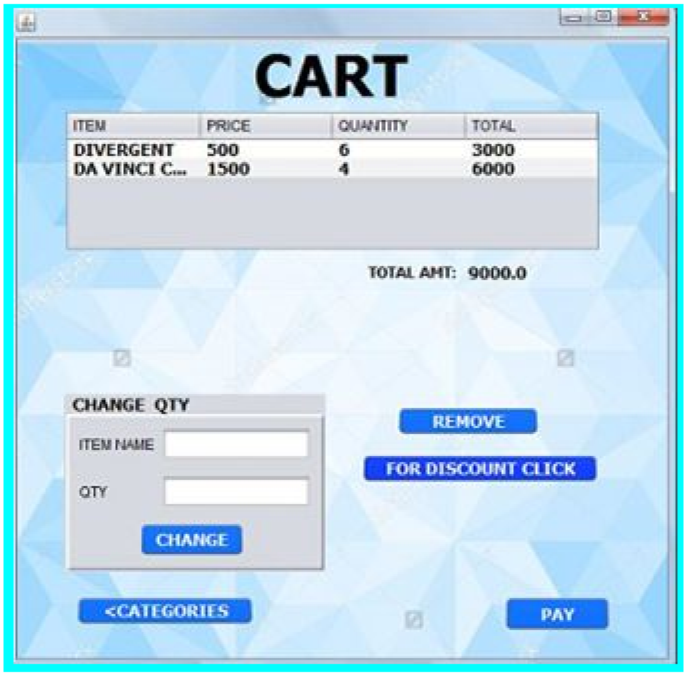
```
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*1500;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Titan
Green',1500,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void add_watch_men3ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*2000;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Titan
Brown',2000,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void back_mwatchActionPerformed(java.awt.event.ActionEvent evt) {
frame_categories.setVisible(true);
frame_watch_men.setVisible(false);
}
```



Frame_watch_women

```
private void watch_women_b1ActionPerformed(java.awt.event.ActionEvent evt)
{JOptionPane.showMessageDialog(null,"Description:Laurell Orange Price:500");}
private void watch_women_b2ActionPerformed(java.awt.event.ActionEvent evt)
{JOptionPane.showMessageDialog(null,"Description:Titan Silver Price:1000");}
private void watch_women_b3ActionPerformed(java.awt.event.ActionEvent evt)
{JOptionPane.showMessageDialog(null,"Description:Titan Blue&Silver Price:1200");}
private void add_watch_women1ActionPerformed(java.awt.event.ActionEvent
evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*500;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Laurell',500,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void add_watch_women2ActionPerformed(java.awt.event.ActionEvent
evt) {
```

```
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*1000;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Titan
Silver',1000,"+qty+","+tot+");");}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}

private void add_watch_women3ActionPerformed(java.awt.event.ActionEvent
evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*1200;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Titan
Blue',5000,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void back_wwatchActionPerformed(java.awt.event.ActionEvent evt) {
frame_categories.setVisible(true);
frame_watch_women.setVisible(false);
}
private void opencart_watch_womenActionPerformed(java.awt.event.ActionEvent
evt) {
frame_cart.setVisible(true);
frame_watch_women.setVisible(false);
}
```



ITEM	PRICE	QUANTITY	TOTAL
DIVERGENT	500	6	3000
DA VINCI C...	1500	4	6000

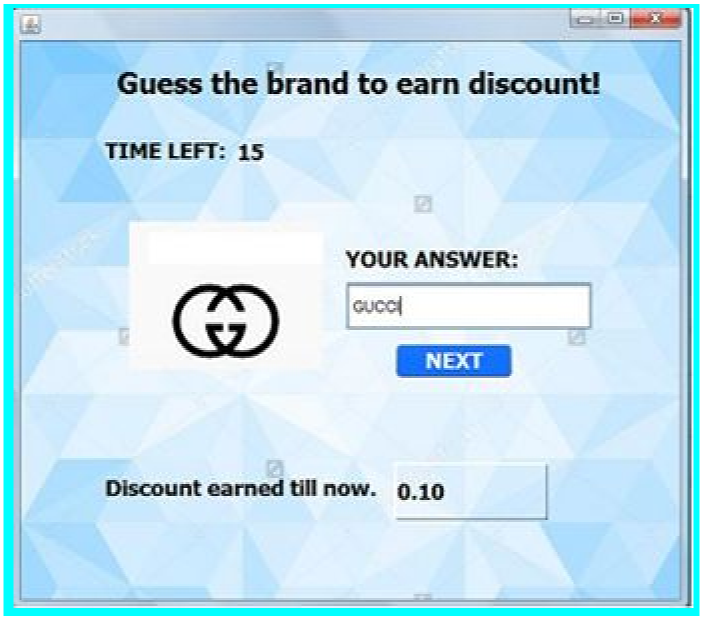
TOTAL AMT: 9000.0

Frame_cart

```
private void frame_cartWindowActivated(java.awt.event.WindowEvent evt){
DefaultTableModel s_cart=(DefaultTableModel)cart.getModel();
s_cart.setRowCount(0);
try{
String sql="select * from "+cart_name+";";
rs=st.executeQuery(sql);
rs.first();
do{
Object[]row={rs.getString(1),rs.getInt(2),rs.getInt(3),rs.getInt(4)};
s_cart.addRow(row);
}while(rs.next());
//A do while loop is used to add the rows from the result set into the jtable.
ResultSet tot=st.executeQuery("select sum(total) from "+cart_name+";");
tot.first();
total=tot.getInt(1);
L1.setText(""+total);
if(total>8000){ discount.setVisible(true);}
else discount.setVisible(false);
//Discount is offered to the user only if the total amount exceeds Rs.8000.
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
```

```
}
private void change_cartActionPerformed(java.awt.event.ActionEvent evt) {
//Changes Quantity of items in cart.User inputs the Item name and desired quantity into
the respective TextFields.
int qty=Integer.parseInt(change_qty.getText());
String it_name=change_name.getText();
String sql1="update "+cart_name+" set qty="+qty+" where item='"+it_name+"';";
//query changes the qty of the corresponding item.
String sql2="update "+cart_name+" set total=price*qty;";
//This query changes the total amount of the item.
String sql3="select * from "+cart_name+";";
try{
DefaultTableModel dtm=(DefaultTableModel)cart.getModel();
dtm.setRowCount(0);
st.executeUpdate(sql1);
st.executeUpdate(sql2);
rs=st.executeQuery(sql3);
rs.first();
do{
Object arr[]={rs.getString(1),rs.getInt(2),rs.getInt(3),rs.getInt(4)};
dtm.addRow(arr);
}while(rs.next());
ResultSet tot=st.executeQuery("select sum(total) from "+cart_name+";");
tot.first();
total=tot.getInt(1);
//a global variable.total amount w/o discount.
L1.setText(""+total);
if(total>8000){
discount.setVisible(true);}
else discount.setVisible(false);
}
catch(Exception e){JOptionPane.showMessageDialog(rootPane,e.getMessage());}
}
private void remove_cartActionPerformed(java.awt.event.ActionEvent evt) {
//Changes Quantity of items in cart.User inputs the Item name and desired quantity into
the respective TextFields.
String name=JOptionPane.showInputDialog(null,"ENTER ITEM NAME","Item Qty",3);
//user inputs the name of the item to be deleted in the InputDialog box.
String sql="delete from "+cart_name+" where item='"+name+"';";
```

```
//deletes the corresponding row from the table in mysql.
try{
DefaultTableModel dtm=(DefaultTableModel)cart.getModel();
dtm.setRowCount(0);
st.executeUpdate(sql);
rs=st.executeQuery("select * from "+cart_name+";");
rs.first();
do{
Object arr[]={rs.getString(1),rs.getInt(2),rs.getInt(3),rs.getInt(4)};
dtm.addRow(arr);
}while(rs.next());
}
catch(Exception e){JOptionPane.showMessageDialog(null,e.getMessage());}
}

private void discountActionPerformed(java.awt.event.ActionEvent evt) {
frame_discount.setVisible(true);
frame_cart.setVisible(false);
}
private void cart_payActionPerformed(java.awt.event.ActionEvent evt) {
frame_bill.setVisible(true);
frame_cart.setVisible(false);
//Takes the user the the frames that displays the details for payment.
}
private void back_cat_cartActionPerformed(java.awt.event.ActionEvent evt) {
frame_categories.setVisible(true);
frame_cart.setVisible(false);
//Takes user back to the categories frame to continue shopping.
}
```



Guess the brand to earn discount!

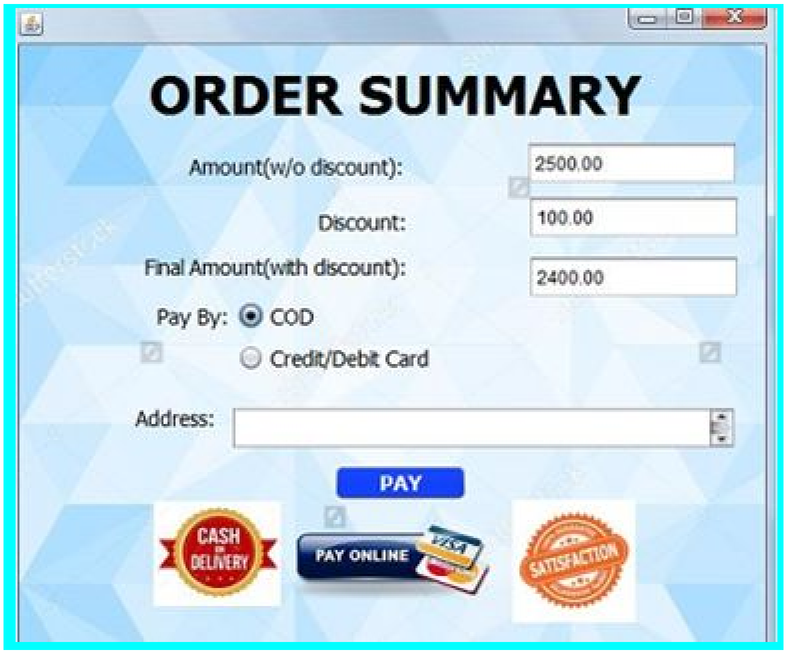
TIME LEFT: 15

YOUR ANSWER: GUCCI

Discount earned till now. 0.10

Frame_discount

```
private void frame_discountWindowOpened(java.awt.event.WindowEvent evt) {
pic_disc.setIcon(new ImageIcon("C:\\IP PROJECT\\game_1.jpg"));
//Displays the 1st pic of the game.
disc_game.start();
//game starts the moment to frame opens.
}
private void nextActionPerformed(java.awt.event.ActionEvent evt) {
String ans=ans_disc.getText();
switch(disc_pic){
case 1:if(ans.equalsIgnoreCase("ucb")){correct++;}break;
case 2:if(ans.equalsIgnoreCase("rado")){correct++;}break;
case 3:if(ans.equalsIgnoreCase("lee")){correct++;}break;
case 4:if(ans.equalsIgnoreCase("monte carlo")){correct++;}break;
case 5:if(ans.equalsIgnoreCase("casio")){correct++;}break;
case 6:if(ans.equalsIgnoreCase("titan")){correct++;}break;
case 7:if(ans.equalsIgnoreCase("rolex")){correct++;}break;
default:}
ans_disc.setText("");
ans_disc.requestFocus();
disc_pic++;
pic_disc.setIcon(new ImageIcon("C:\\IP PROJECT\\game_"+disc_pic+".jpg"));
disc=correct*0.1;
disc_disc.setText(""+disc+"%");
//Every time the user clicks the next button,a new image is displayed on the label.The images are named in a serial order so that number of the picture corresponds to the number in the switch case.This is achieved using a global variable that is incremented with every click.
}
```



ORDER SUMMARY

Amount(w/o discount): 2500.00

Discount: 100.00

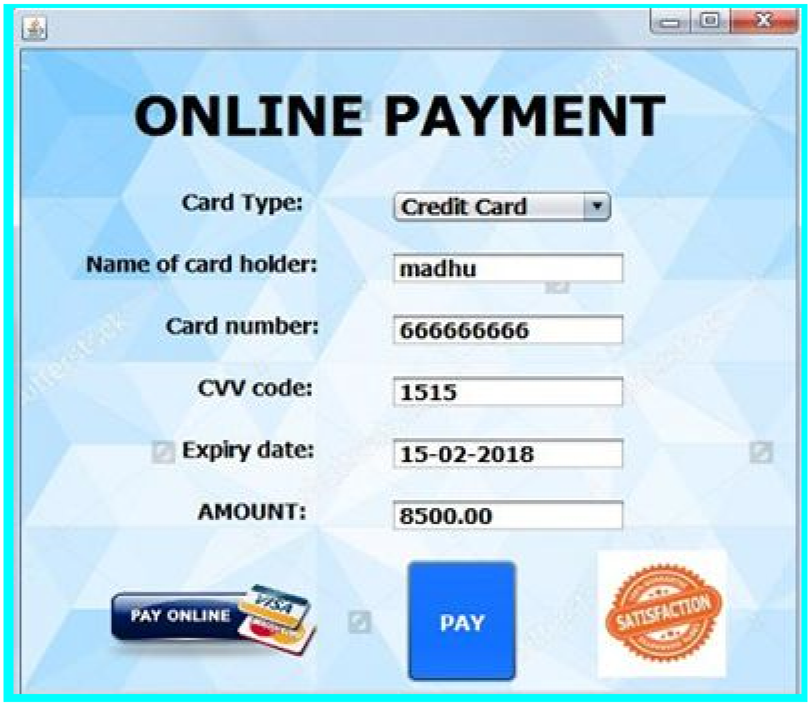
Final Amount(with discount): 2400.00

Pay By: ● COD ○ Credit/Debit Card

Address:

Frame_bill

```
private void frame_billWindowOpened(java.awt.event.WindowEvent evt) {
item_sum.setText(""+total);
tf_disc.setText(""+disc);
fin_amt=total-(total*disc);
tf_tot_amt.setText(""+fin_amt);
}
private void pay_billActionPerformed(java.awt.event.ActionEvent evt) {
String add=ta_add.getText();
String boo="";
if(add.equals(boo)){JOptionPane.showMessageDialog(rootPane,"PLEASE ENTER ADDRESS");}
if (rb_cod.isSelected()){frame_cod.setVisible(true);
frame_bill.setVisible(false);}
else if (rb_credit.isSelected()){frame_credit.setVisible(true);
frame_bill.setVisible(false);}
else JOptionPane.showMessageDialog(null,"choose a method of payment");
//The user is given two payment options-cash on delivery,credit/debit card.Addition of a buttongroup has made the two options exclusive.
}
```



ONLINE PAYMENT

Card Type: Credit Card

Name of card holder: madhu

Card number: 666666666

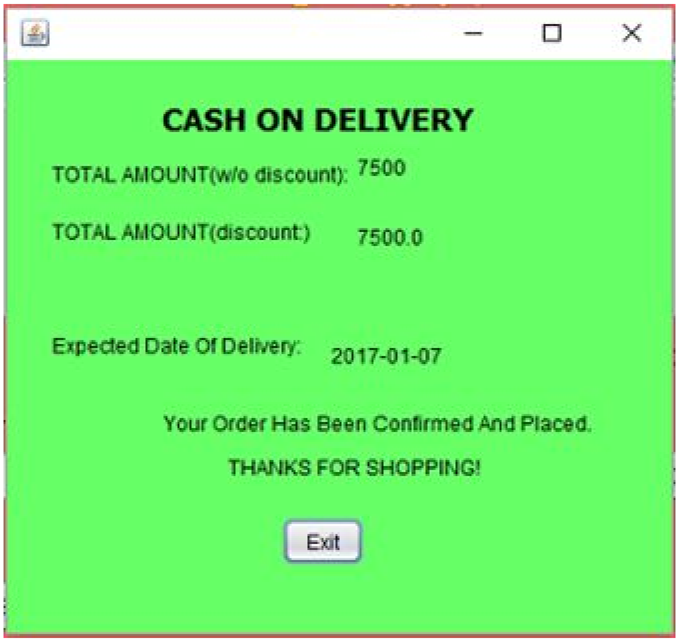
CVV code: 1515

Expiry date: 15-02-2018

AMOUNT: 8500.00

Frame_credit

```
private void frame_creditWindowOpened(java.awt.event.WindowEvent evt) {
lbl_c_amt.setText(""+fin_amt);
try{
String sql1="use shoppingcart;";
String sql2="create table credit(name varchar(40),c_no int(20),cvv int(20),e_date date,amt decimal(20,10));";
st.executeUpdate(sql1);
st.executeUpdate(sql2);
//A table credit is created in case it doesn't already exist in the database.
}
catch(Exception e){System.out.println("");}
}
private void pay_creditActionPerformed(java.awt.event.ActionEvent evt) {
String name=name_c.getText();
int cardno=Integer.parseInt(cardno_c.getText());
int cvv=Integer.parseInt(cvv_c.getText());
String date=expd_c.getText();
Float amt=Float.parseFloat(amt_c.getText());
String sql="insert into credit values('"+name+"',"+cardno+","+cvv+",'"+date+"',"+amt+");";
try{
st.executeUpdate(sql);
//The credit/debit card information input by the user is stored in the table credit.
JOptionPane.showMessageDialog(null,"PAYMENT COMPLETED.THANKS FOR SHOPPING.");
System.exit(0);
}
catch(Exception e){JOptionPane.showMessageDialog(null,e.getMessage());}
}
```



CASH ON DELIVERY

TOTAL AMOUNT(w/o discount): 7500

TOTAL AMOUNT(discount:) 7500.0

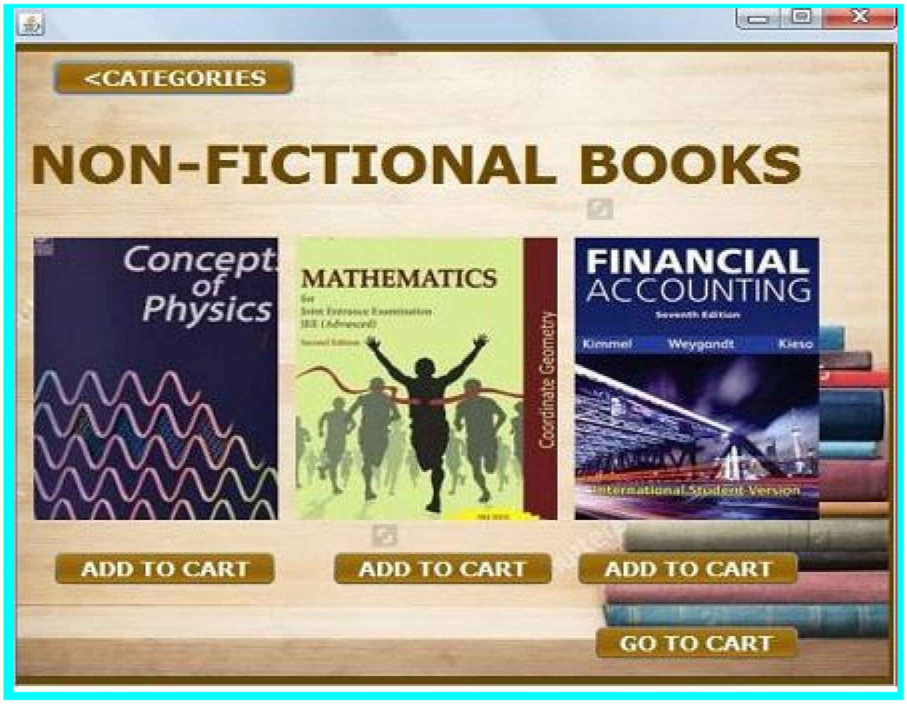
Expected Date Of Delivery: 2017-01-07

Your Order Has Been Confirmed And Placed.

THANKS FOR SHOPPING!

Frame_cod

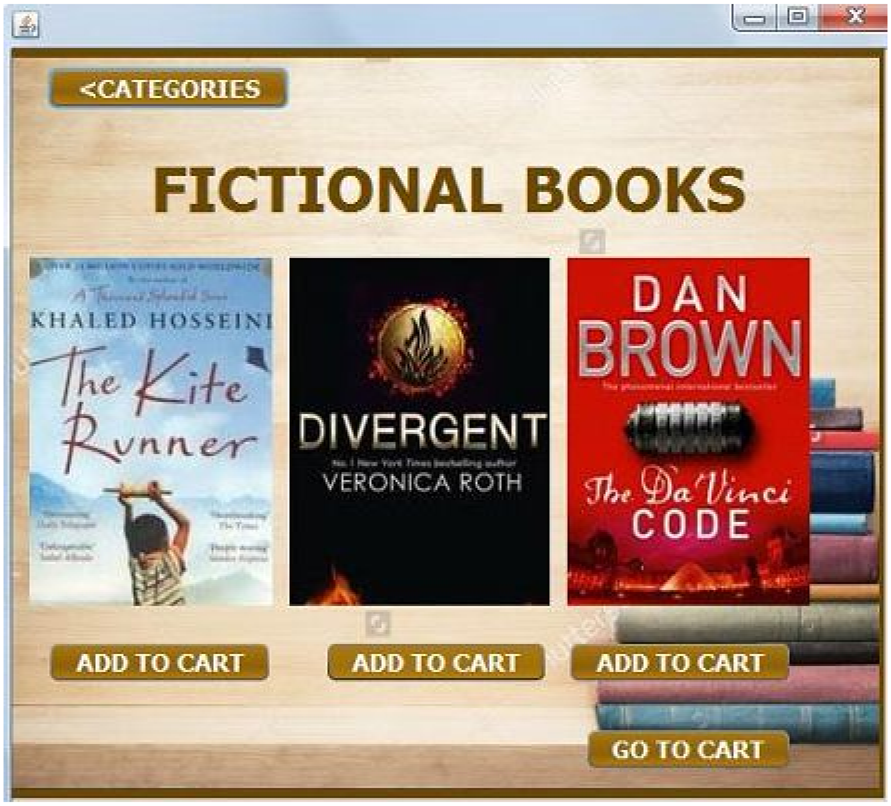
```
private void frame_codWindowActivated(java.awt.event.WindowEvent evt){
lbl_amt.setText(""+total);
lbl_disc_amt.setText(""+fin_amt);
//Total & fin_amt are global variables.
try{
String sql1="select date_add(curdate(),interval 5 day);";
ResultSet r=st.executeQuery(sql1);
r.first();
String d=(r.getString(1));
System.out.println(d);
date_label.setText(""+d);
//The user is given a tentative date to expect the delivery of the items.
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void exit_codActionPerformed(java.awt.event.ActionEvent evt) {
System.exit(0);
}
```



Frame_nonfic

```
private void book_nonfic_b1ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:HCV Price:Rs.200");
}
private void book_nonfic_b2ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Cengage Math Price:Rs.300");
}
private void book_nonfic_b3ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Accounting 7th edition. Price:500");
}
private void add_nonficb1ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item Qty",3));
int tot=qty*200;
total_amt=total_amt+tot;
try{
```

```
st.executeUpdate("insert into "+cart_name+" values ('hcv',200,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void add_ nonficb2 ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*300;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('cengage',300,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void add_non ficb3ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*500;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Accounts',500,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void back_non ficActionPerformed(java.awt.event.ActionEvent evt) {
frame_categories.setVisible(true);
frame_nonfic.setVisible(false);
}
private void opencart_nonfic ActionPerformed(java.awt.event.ActionEvent evt) {
frame_cart.setVisible(true);
frame_nonfic.setVisible(false);
}
```



Frame_fictional

```
private void book_fic_b1ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Kite Runner Price:Rs.350");
}
private void book_fic_b2ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Divergent Price:Rs.350");
}
private void book_fic_b3ActionPerformed(java.awt.event.ActionEvent evt) {
JOptionPane.showMessageDialog(null,"Description:Da Vinci Code Price:Rs.350");
}
private void add_ficb1ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*350;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Kite
Runner',350,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void add_ ficb2 ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*350;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Divergent',350,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void add_ ficb3ActionPerformed(java.awt.event.ActionEvent evt) {
int qty=Integer.parseInt(JOptionPane.showInputDialog(null,"Enter Quantity ","Item
Qty",3));
int tot=qty*350;
total_amt=total_amt+tot;
try{
st.executeUpdate("insert into "+cart_name+" values ('Da Vinci
Code',350,"+qty+","+tot+");");
}
catch(Exception e) {JOptionPane.showMessageDialog(null,e.getMessage());}
}
private void back_ ficActionPerformed(java.awt.event.ActionEvent evt) {
frame_categories.setVisible(true);
frame_fictional.setVisible(false);
}
private void opencart_fic ActionPerformed(java.awt.event.ActionEvent evt) {
frame_cart.setVisible(true);
frame_fictional.setVisible(false);
}
```