ACKNOWLEDGEMENT:-

I would like to express my greatest gratitude to the people who have helped and supported me throughout my project. I am grateful to my teacher, **MR BHARAT RANA** for his continuous support for the project from initial advice and encouragement till this day.

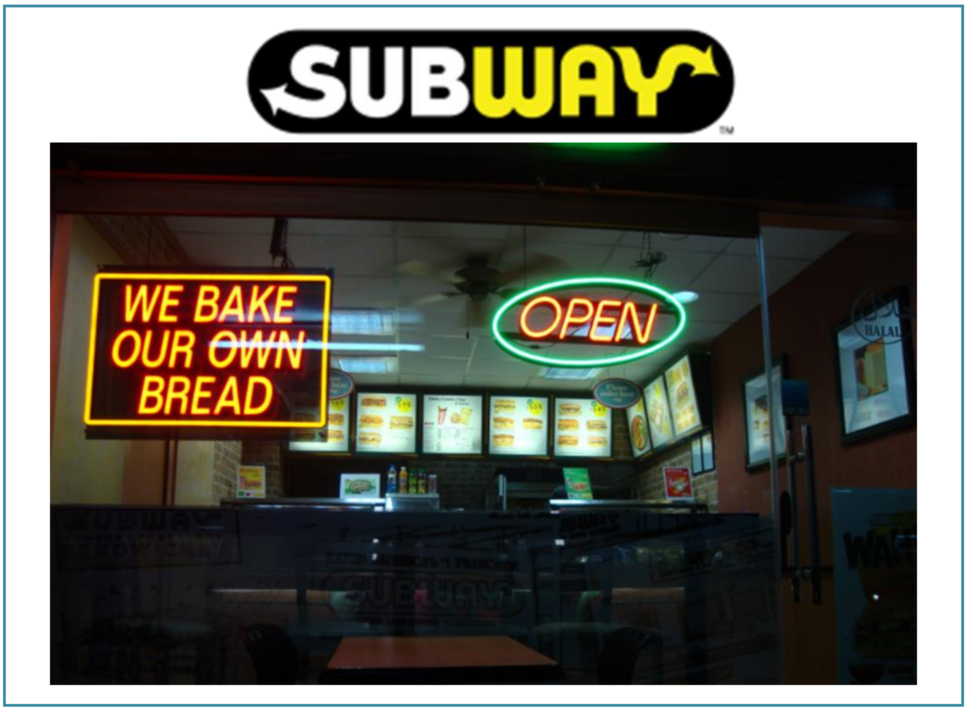
A special thank goes to my group partner who helped me in completing this project by providing their interesting ideas and thoughts which made this project easier and accurate.

Also I would like to express my gratitude towards the school authority and also my parents due to which we were able to complete the project successfully.

CONTENTS:-

ABOUT SUBWAY:-

SUBWAY AT DAMAN



SUBWAY is an American fast food restaurant franchise that primarily sells submarine sandwiches and salads. It is owned and operated by Doctor's Associates, Inc. Subway is one of the fastest growing franchises in the world with 40,735 restaurants in 102 countries and territories. It is the largest single-brand restaurant chain and the largest restaurant operator globally. Subway's main operations office is in **Milford, Connecticut**.

SUBWAY'S HISTORY:-

SUBWAY
TM
SUBWAY
WAH!
₹50*
all day, every day.
spicy
aloo patty sub
recommended with chilli sauce
Terms and Conditions
YOU CAN MAKE A DIFFERENCE
Own a Franchise
How To Order
Different Sub, Every Day
Get News & Offers
Tell Subway
HOME | MENU/NUTRITION | REQUIRE REAL ESTATE | ABOUT US | RESTAURANT LOCATOR |
CONTACT US | TERMS OF USE | PARTNERS™ | PRIVACY POLICY | SIGN UP | UNSUBSCRIBE| SITEMAP
Like box
Find us

Take our customer experience survey and be rewarded with a free cookie*

SUBWAY® is a registered trademark of Doctor's Associates Inc.
2013 Doctor's Associates Inc. All rights reserved.

In 1965, Fred DeLuca starts "Pete's Super Submarines" in Bridgeport, Connecticut, and in the following year they formed Doctor's Associates Inc to oversee operations of the restaurants as the franchise expanded. In 1968, the shop began using the name "Subway" for the first time. The first Subway on West Coast was opened in Fresno, California in 1978. The first Subway outside of North America opened in Bahrain, in December 1984. In 2004, Subway began opening stores in Walmart supercenters. Subway's best-selling sandwich the B.M.T., originally stood for Brooklyn Manhattan Transit but now stands for "Biggest, Meatiest and Tastiest". Subway uses the advertising slogan "Eat Fresh".

NEED FOR THE SYSTEM:-

- It is user friendly, needs no training to operate.
- The chance of committing mistakes while entering data in the computer is almost impossible.
- The system analyzes Subway's Transactions and automatically provides a statistical data of Subway's Scale and Profit.
- The System reduces the man power thus saving restaurant's money and prevents overcrowding of the attendees.
- The System takes care of the Security of the data most efficiently.
- The System reduces the chances of redundancy in accounts and manages the account accurately.

FEASIBILITY:-

➢ TECHNICAL FEASIBLITY:

Technical feasibility is considered with the technical requirements for the system.

Weather the software and hardware required for the system is available with the

Organization or can be obtained with ease.

➢ HARDWARE FEASIBLITY:

The proposed system will need an Intel i-7 processor and HP LaserJet color printer.

➢ SOFTWARE REQUIREMENTS:

The proposed system will be developed in the JAVA database package. And fortunately the JAVA is available with the organization.

➢ ECONOMICAL FEASIBLITY:

Development of the system can be fruitful if it provides cost effectiveness i.e. whatever cost is supposed to be invested for the purposed system for a period of approval.

➢ MAINTENANCE:

The proposed system to workable with its effectiveness for at least 5 years we consider that 10% of cost will be cost of the hardware per annum. Also the consumable item like stationary would cost 2000 per annum. The maintenance of the system can be handled by one EDP.

➢ TRAINING:

As the proposed system is quite user friendly and menu driven such that normal man can easily grasp the operation procedures.

➢ INTANGIBLE BENEFITS:

Apart from the tangible benefits the intangible benefits are always the real motivation behind any system development. Because for any modern organization money is not that much important as the professional work environment and proper MIS system.

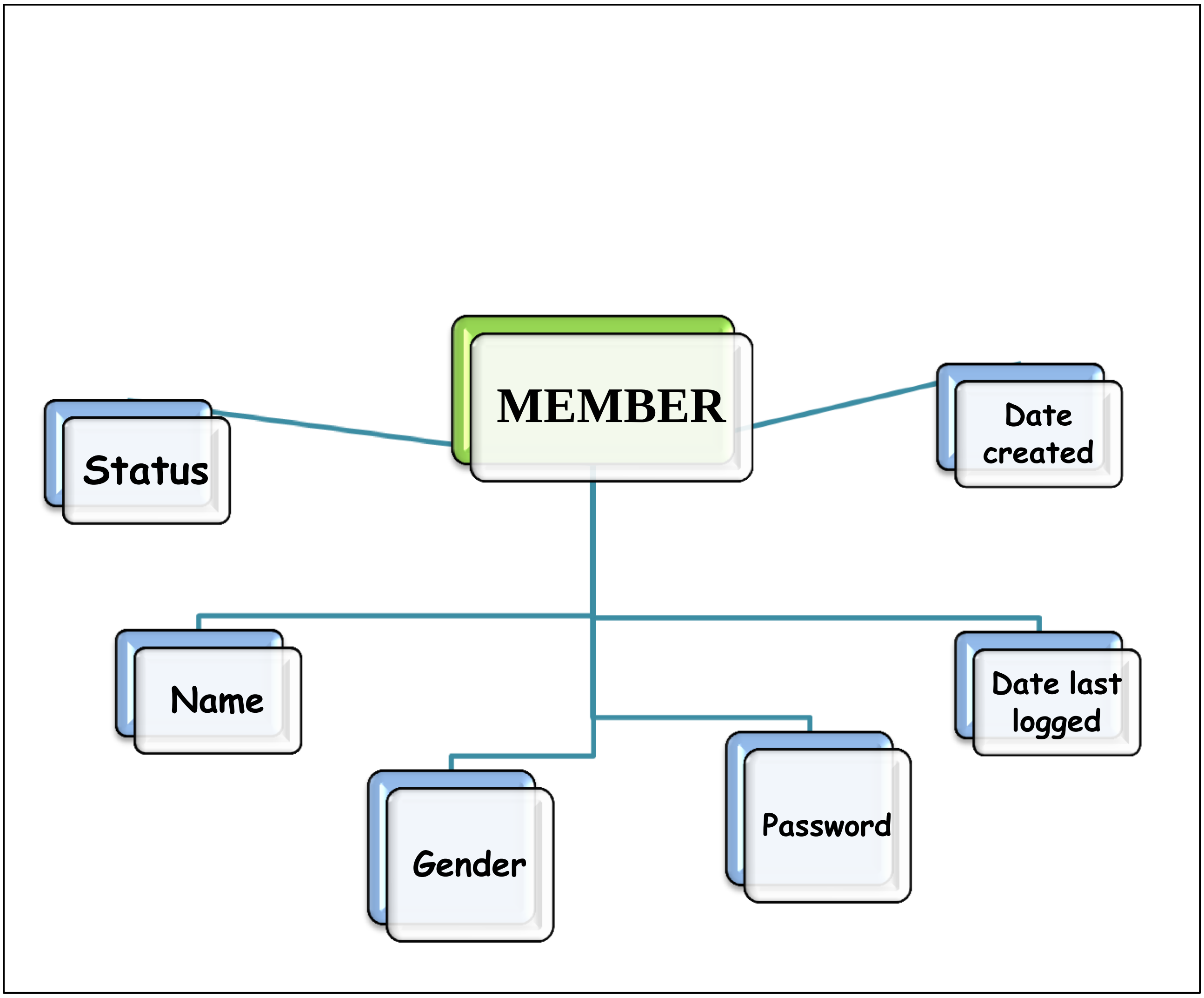
OBJECTIVES:-

- To allow more than one user work per day and on each data.
- To reduce the usage of the manpower.
- To bring accuracy to your management.
- To allow fast accessing of data.
- To make the system more friendly to native use.
- To avoid data redundancy and inconsistency.
- To reduce the load of calculation and accounts.
- To provide a comfortable platform for customers for enquiry and personal purchasing.

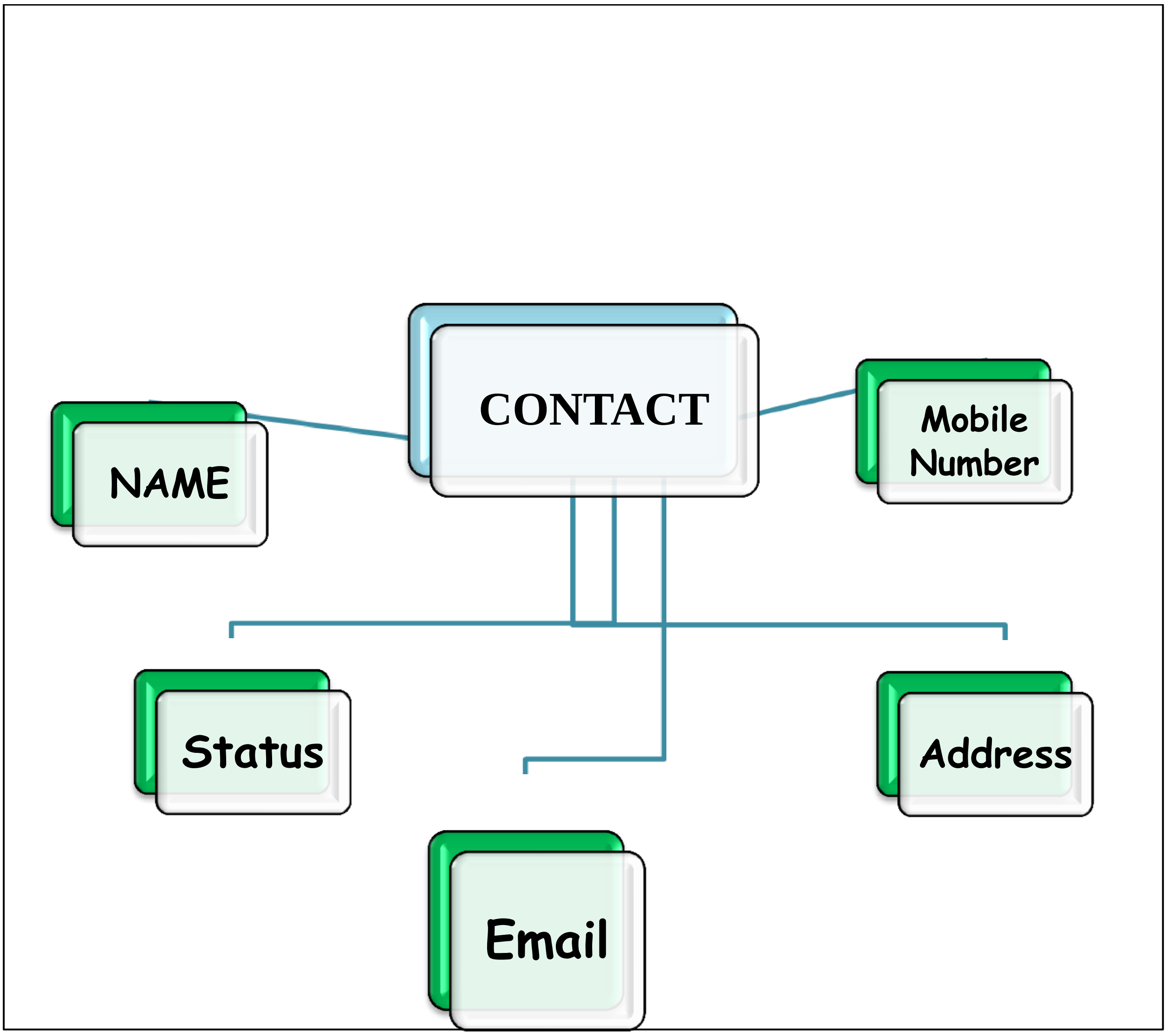
DATABASE DESIGN:-

TABLE STRUCTURES

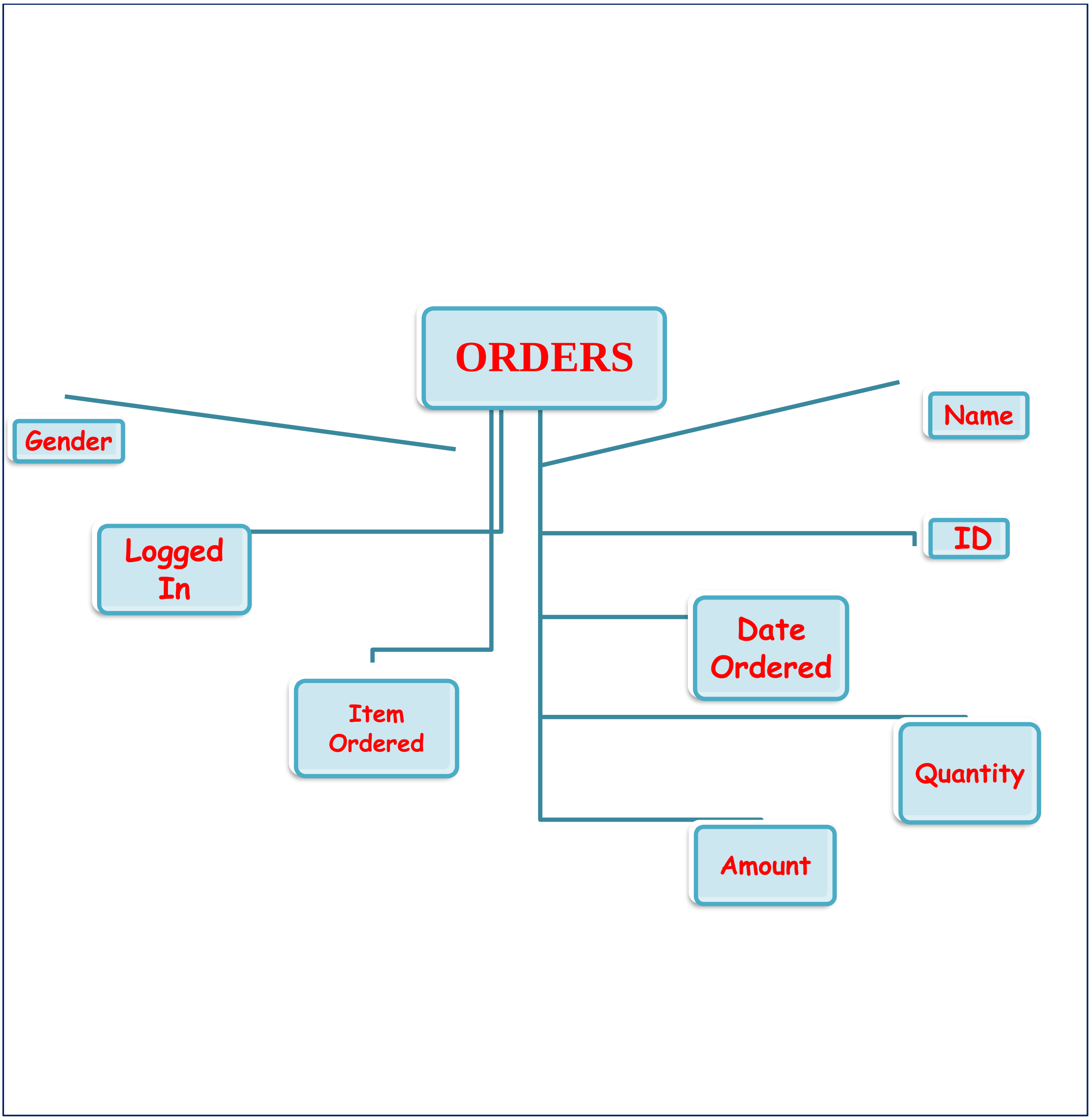
DESCRIPTION: - Details of Members



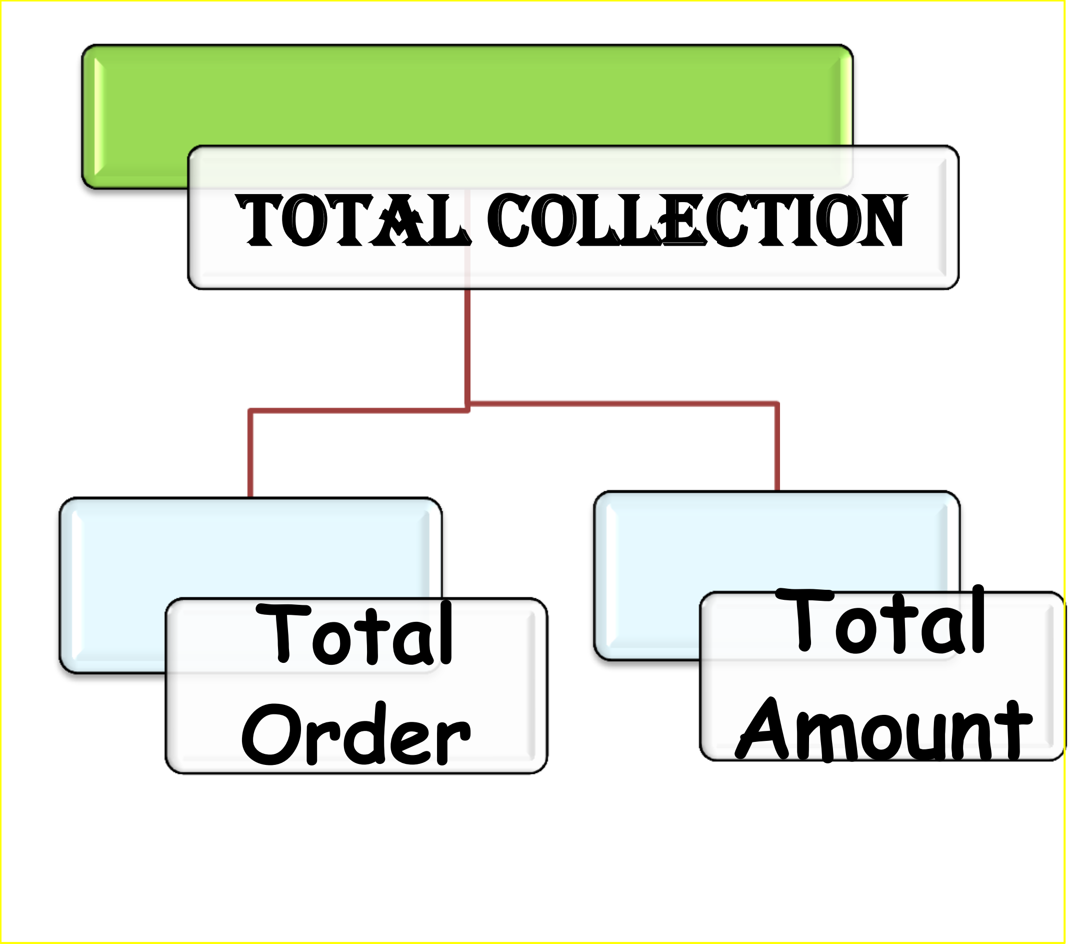
DESCRIPTION:- Contact Details



DESCRIPTION: - Order Details



DESCRIPTION: - Details of Total Collection



DESCRIPTION: - Details of Total Collection Per day.

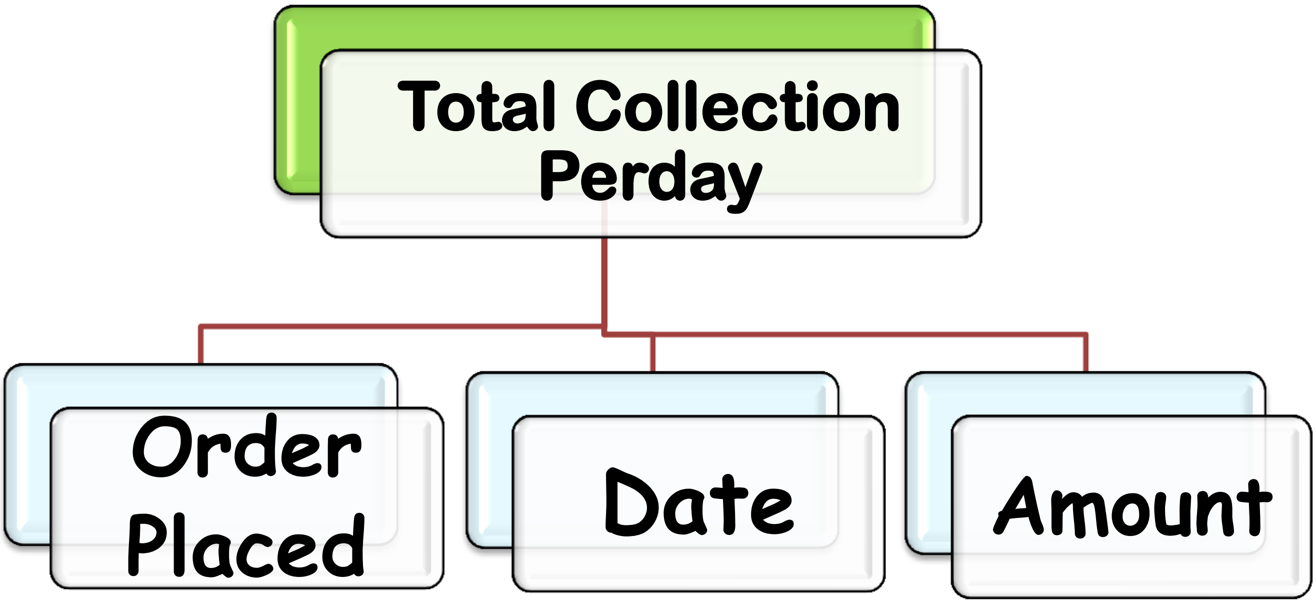
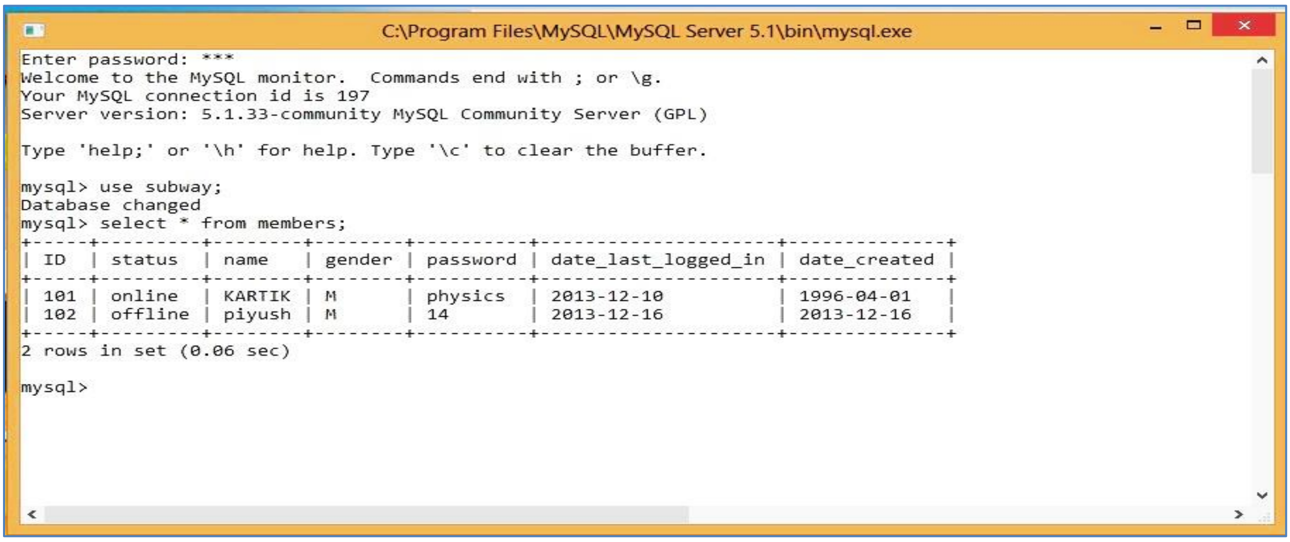


TABLE STRUCTURE

Table: - Member



```
Enter password: ***
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 197
Server version: 5.1.33-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use subway;
Database changed
mysql> select * from members;
+-----+---------+--------+--------+----------+----------------------+---------------+
| ID  | status  | name   | gender | password | date_last_logged_in  | date_created  |
+-----+---------+--------+--------+----------+----------------------+---------------+
| 101 | online  | KARTIK | M      | physics  | 2013-12-10           | 1996-04-01    |
| 102 | offline | piyush | M      | 14       | 2013-12-16           | 2013-12-16    |
+-----+---------+--------+--------+----------+----------------------+---------------+
2 rows in set (0.06 sec)

mysql>
```

MySQL Command:-

create table members

(ID integer(5) primary key,

status varchar(7) not null,

name varchar(30) not null,

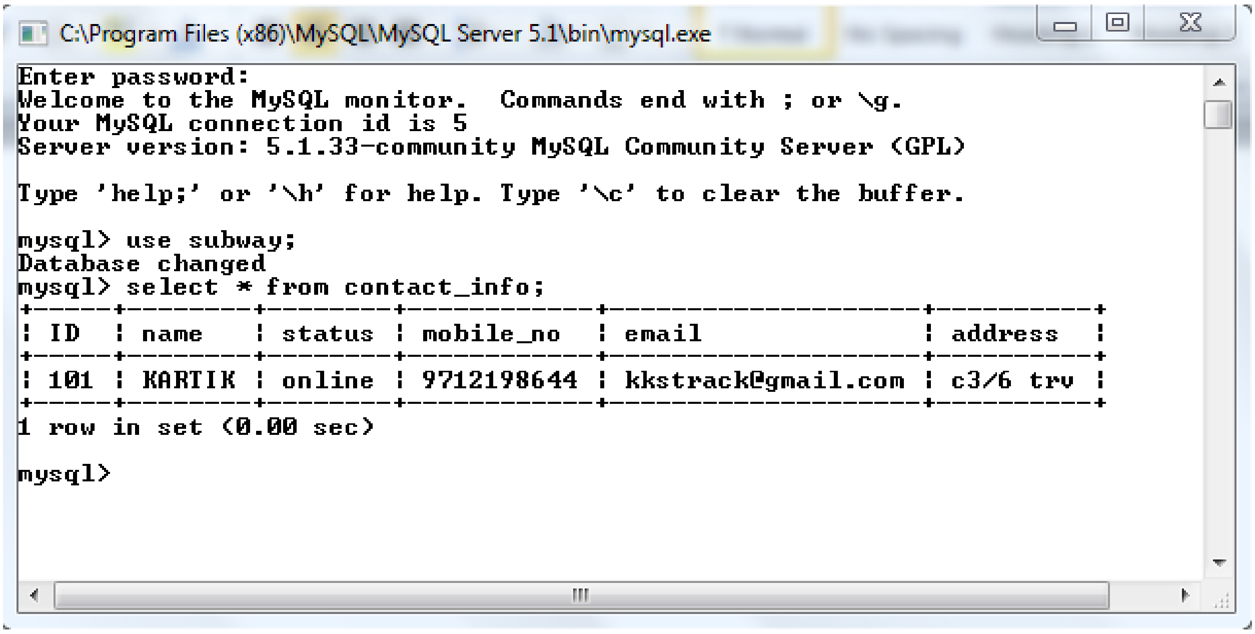
gender char(1) not null,

password varchar(30) not null,

date_last_logged_in date not null,

date_created date not null);

Table: - Contact Info

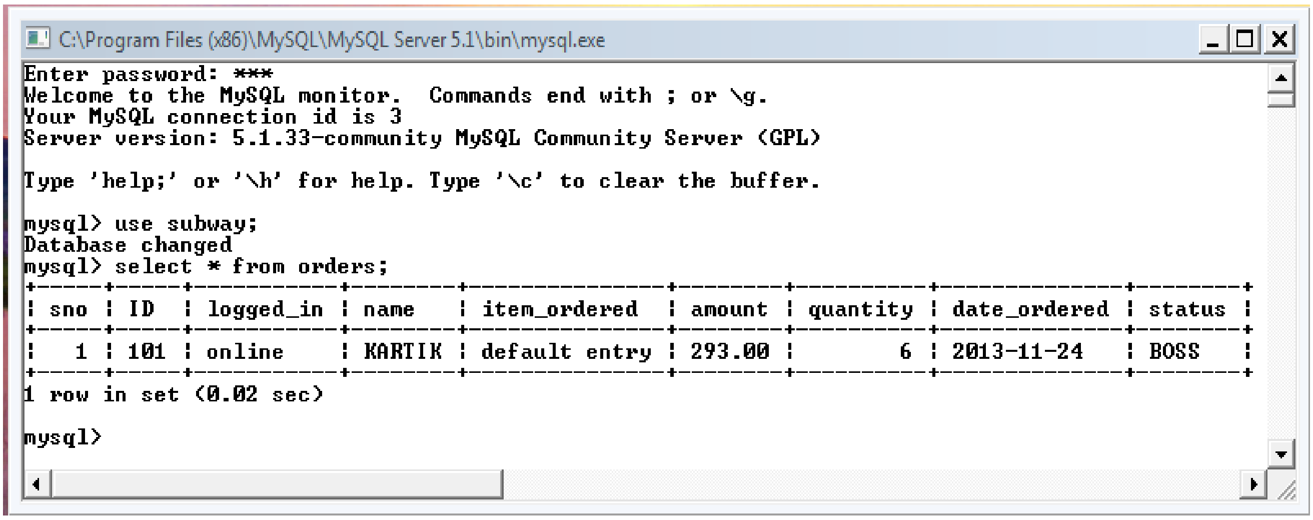


```
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 5
Server version: 5.1.33-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use subway;
Database changed
mysql> select * from contact_info;
+-----+--------+--------+------------+---------------------+-----------+
| ID  | name   | status | mobile_no  | email               | address   |
+-----+--------+--------+------------+---------------------+-----------+
| 101 | KARTIK | online | 9712198644 | kkstrack@gmail.com  | c3/6 trv  |
+-----+--------+--------+------------+---------------------+-----------+
1 row in set (0.00 sec)

mysql>
```

MySQL Command:-

```
create table contact_info

(ID integer(5) primary key,

name varchar(30) not null,

status varchar(7) not null,

mobile_no bigint(10) not null,

email varchar(40) not null,

address varchar(50) not null);
```

Table: - Orders



```
Enter password: ***
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3
Server version: 5.1.33-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use subway;
Database changed
mysql> select * from orders;
+-----+-----+-----------+--------+---------------+--------+----------+--------------+--------+
| sno | ID  | logged_in | name   | item_ordered  | amount | quantity | date_ordered | status |
+-----+-----+-----------+--------+---------------+--------+----------+--------------+--------+
|   1 | 101 | online    | KARTIK | default entry | 293.00 |        6 | 2013-11-24   | BOSS   |
+-----+-----+-----------+--------+---------------+--------+----------+--------------+--------+
1 row in set (0.02 sec)

mysql>
```

MySQL Command:-

create table orders

(sno integer(5) not null,

ID integer(5) not null,

logged_in varchar(7),

name varchar(30) not null,

item_ordered varchar(50) not null,

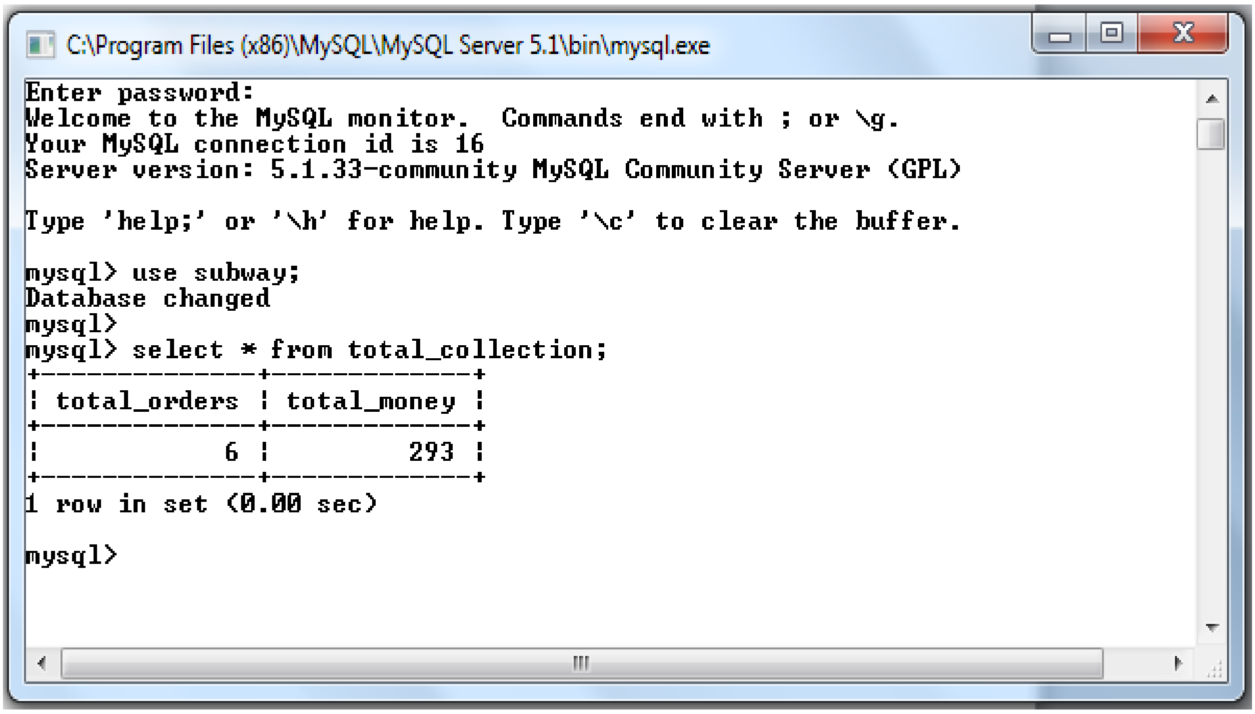
amount decimal(8,2) not null,

quantity integer(5) null,

date_ordered date,

status varchar(6) not null);

Table: - Total Collection

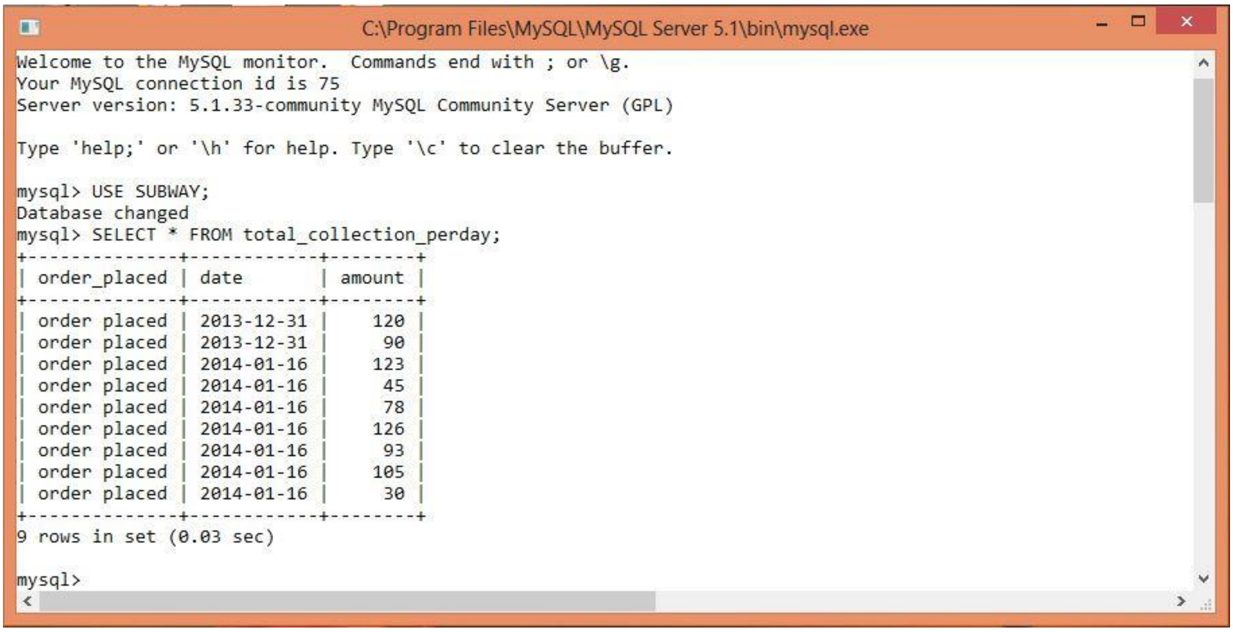


```
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 16
Server version: 5.1.33-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> use subway;
Database changed
mysql>
mysql> select * from total_collection;
+--------------+-------------+
| total_orders | total_money |
+--------------+-------------+
|            6 |         293 |
+--------------+-------------+
1 row in set (0.00 sec)

mysql>
```

MySQL Command:-

```
create table total_collection
(total_orders integer(8),
total_money integer(8));
```

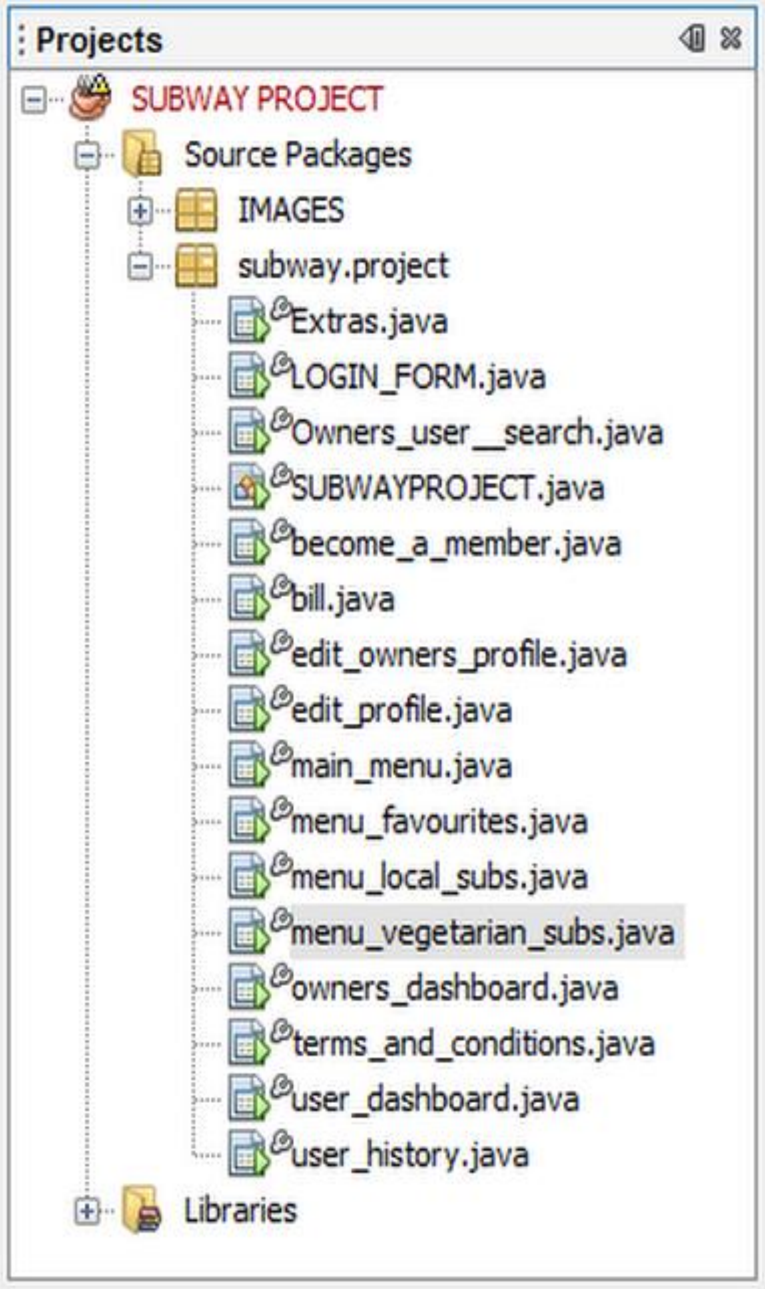
TABLE: - TOTAL COLLECTION PERDAY



```
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 75
Server version: 5.1.33-community MySQL Community Server (GPL)

Type 'help;' or '\h' for help. Type '\c' to clear the buffer.

mysql> USE SUBWAY;
Database changed
mysql> SELECT * FROM total_collection_perday;
+--------------+------------+--------+
| order_placed | date       | amount |
+--------------+------------+--------+
| order placed | 2013-12-31 |    120 |
| order placed | 2013-12-31 |     90 |
| order placed | 2014-01-16 |    123 |
| order placed | 2014-01-16 |     45 |
| order placed | 2014-01-16 |     78 |
| order placed | 2014-01-16 |    126 |
| order placed | 2014-01-16 |     93 |
| order placed | 2014-01-16 |    105 |
| order placed | 2014-01-16 |     30 |
+--------------+------------+--------+
9 rows in set (0.03 sec)

mysql>
```

```
create table total_collection_perday

(order_placed varchar(15),

date date,

amount integer(8));
```

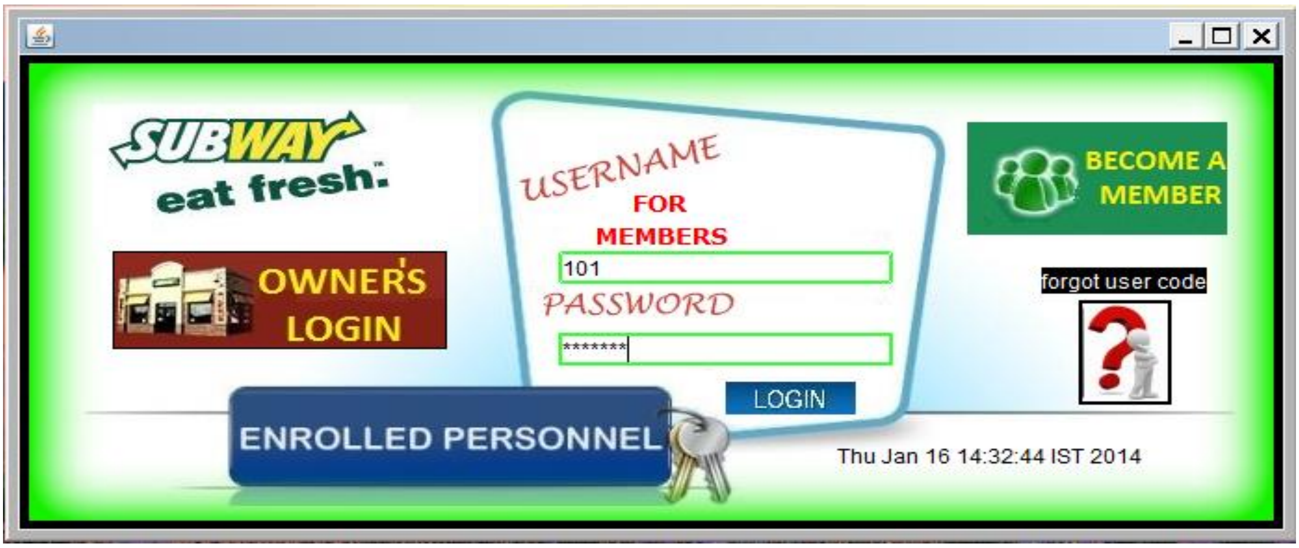
PROJECT EXPLORER



Projects

```
SUBWAY PROJECT
  Source Packages
    IMAGES
    subway.project
      Extras.java
      LOGIN_FORM.java
      Owners_user__search.java
      SUBWAYPROJECT.java
      become_a_member.java
      bill.java
      edit_owners_profile.java
      edit_profile.java
      main_menu.java
      menu_favourites.java
      menu_local_subs.java
      menu_vegetarian_subs.java
      owners_dashboard.java
      terms_and_conditions.java
      user_dashboard.java
      user_history.java
  Libraries
```

Main Frame: Login Form



101

****

Thu Jan 16 14:32:44 IST 2014

SOURCE CODE: Login Form.java

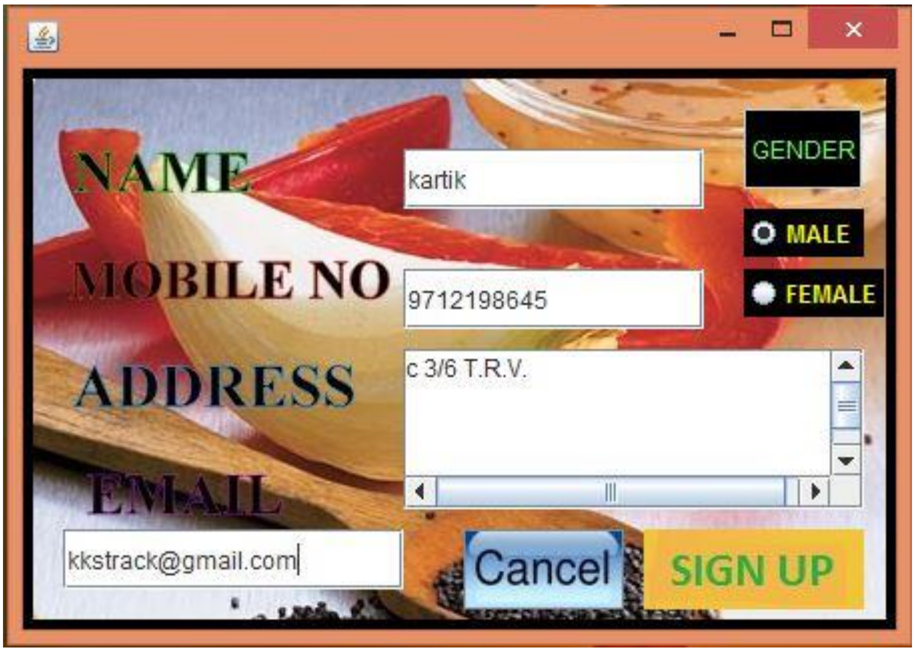
```
import java.sql.*;

import javax.swing.ImageIcon;

import javax.swing.JOptionPane;


private void loginBtnActionPerformed(java.awt.event.ActionEvent evt) {


String ID=usernameTf.getText().trim().toLowerCase();

String pwd=new String(pwdTf.getPassword());
```

```
try {
        Class.forName("java.sql.Driver");
Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/subway","root","123");
Statement stmt=con.createStatement();
String check_ID="select password from members where ID='"+ID+"';";
ResultSet rs=stmt.executeQuery(check_ID);
while(rs.next()){
   String pwd_from_mysql=rs.getString("password");


   if(pwd.equals(pwd_from_mysql)){
 String set_offline="update members set status='offline';";
 String set_offline1="update contact_info set status='offline';";
 String set_offline2="update orders set logged_in='offline';";


String set_online="update members set status='online' where ID="+ID;
String set_online1="update contact_info set status='online' where ID="+ID;
String set_online2="update orders set logged_in='online' where
ID="+ID;

Calendar c = Calendar.getInstance();

String set_date="update members set
date_last_logged_in='"+(c.get(Calendar.YEAR))+"-
"+(c.get(Calendar.MONTH)+1)+"-"+(c.get(Calendar.DATE))+"' where
ID="+ID+" ;";


stmt1.executeUpdate(set_offline);

stmt2.executeUpdate(set_offline1);

stmt6.executeUpdate(set_offline2);


stmt3.executeUpdate(set_online);

stmt4.executeUpdate(set_online1);

stmt5.executeUpdate(set_online2);

stmt7.executeUpdate(set_date);

new user_dashboard().setVisible(true);

dispose();

}


else{

   JOptionPane.showMessageDialog(null, "Incorrect password. Please
try again.");
```

```
    pwdTf.setText("");
      }
}
rs.close();
stmt.close();
con.close();
}
catch(Exception e){
            JOptionPane.showMessageDialog(this,e.getMessage());
}
    }

private void
becomememberBtnActionPerformed(java.awt.event.ActionEvent evt) {
  new terms_and_conditions().setVisible(true);
  dispose();
}

 private void fgtpwdBtnActionPerformed(java.awt.event.ActionEvent evt) {
  String mobile=(JOptionPane.showInputDialog(null, "enter your mobile
number that you entered during sign up!")).toLowerCase().trim();
```

```
try{

Class.forName("java.sql.Driver");

Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/subway","root","123"
);

Statement stmt=con.createStatement();

String query="select ID from contact_info where
mobile_no='"+mobile+"';";

ResultSet rs=stmt.executeQuery(query);

   while(rs.next()){

      String mysql_username=rs.getString("ID");

      JOptionPane.showMessageDialog(null,"your username is:
"+mysql_username);

   }

}

catch (Exception e) {


    }

}

   private void usernameTfKeyTyped(java.awt.event.KeyEvent evt)
{
   char ch= evt.getKeyChar();
   if( ((ch>='a') && (ch<='z')) || ((ch>='A') && (ch<='Z')) ){
       evt.consume();
    }
 }

   private void formWindowGainedFocus(java.awt.event.WindowEvent evt)
{
  Calendar c = Calendar.getInstance();
 time.setText("" +c.getTime());
}
private void owner loginBtnActionPerformed(java.awt.event.ActionEvent evt){
try {
   Class.forName("java.sql.Driver");
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/subway","root"
,"123");
Statement stmt = con.createStatement();
String get_owner_password="select password from members where
ID=101 ;";
ResultSet rs = stmt.executeQuery(get_owner_password);
while(rs.next()){
    String username_mysql=(rs.getString("password"));
  String pwd = JOptionPane.showInputDialog(null, "PLEASE ENTER PASSWORD - password is "+ username_mysql );

if(pwd.equals(username_mysql)){

        new owners_dashboard().setVisible(true);

  dispose();

}

else

    JOptionPane.showMessageDialog(null, "SORRY WRONG PASSWORD- password is "+ username_mysql);

   }

}

    catch(Exception e){

    }

 }
```

Main Frame: Become A Member Form



NAME: kartik

MOBILE NO: 9712198645

ADDRESS: c 3/6 T.R.V.

EMAIL: kkstrack@gmail.com

GENDER: ○ MALE ● FEMALE

SOURCE CODE: Become A Member.java

```
import javax.swing.JOptionPane;

import java.sql.*;

import java.util.Calendar;


private void signupBtnActionPerformed(java.awt.event.ActionEvent evt)
{


String name = nameTf.getText().trim().toLowerCase();

long mobile = Long.parseLong(mobnoTf.getText());
```

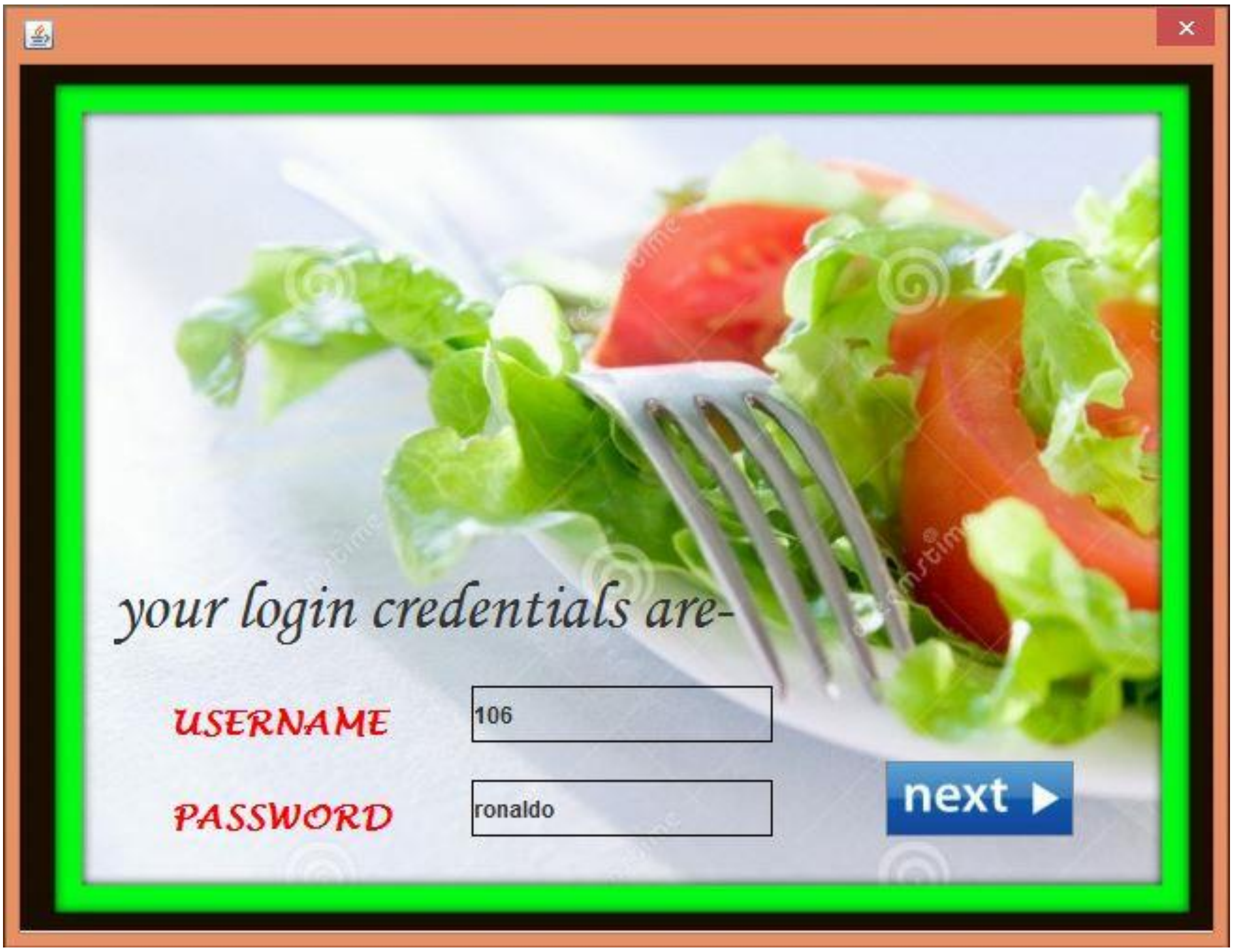
```
String address = addressTa.getText().trim().toLowerCase();

String email_id = emailTf.getText().trim().toLowerCase();

String password = " ";

String password1 = JOptionPane.showInputDialog("please select a password for your account");

String password2 =JOptionPane.showInputDialog("re-enter password to confirm");

if(password1.equals(password2)){

   password=password1;

}

else{

   JOptionPane.showMessageDialog(null, "the password you entered didn't matched, please try again");

   signupBtn.doClick();

}

if(name.equals("")||mobnoTf.getText().equals("")||password.equals("")||email_id.equals("")||address.equals(""))

   JOptionPane.showMessageDialog(null, "All fields are mandatory.");

else{

     try {

         Class.forName("java.sql.Driver");

 Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/subway","root","123");
 Statement stmt = con.createStatement();
 String id = "select ID from members order by ID;";
 ResultSet rs = stmt.executeQuery(id);
 while(rs.next()){
 int new_id = Integer.parseInt(rs.getString("ID"))+1;
 idLb.setText(""+new_id);
 usernameLb.setText(""+new_id);
 pwdLb.setText(""+password);
}
 String gender = " ";
   if(maleRb.isSelected())
    gender="M";
else
     gender="F";

Calendar c = Calendar.getInstance();
String add_member = "insert into members
values("+idLb.getText().trim()+",'offline','"+name+"','"+gender+"','"+p
assword+"','"+(c.get(Calendar.YEAR))+"-
"+(c.get(Calendar.MONTH)+1)+"-
"+(c.get(Calendar.DATE))+"','"+(c.get(Calendar.YEAR))+"-
"+(c.get(Calendar.MONTH)+1)+"-"+(c.get(Calendar.DATE))+"');" ;
```

```
 String update_contact_info="insert into contact_info values("+idLb.getText().trim()+",'"+name+"','offline',"+mobile+",'"+email_id+"','"+address+"');";

  stmt.executeUpdate(add_member);

  stmt.executeUpdate(update_contact_info);

  rs.close();

  stmt.close();

  con.close();

  congratsMsg.setVisible(true);

  dispose();

}

catch(Exception e){

  JOptionPane.showMessageDialog(this,e.getMessage());

  new become_a_member().setVisible(true);

  dispose();

   }

 }

}

    private void mobnoTfFocusLost(java.awt.event.FocusEvent evt)
{

 String mobile=mobnoTf.getText().trim();

 if(mobile.equals("")){

}
```

```
else{
    int initial_digit=Integer.parseInt(mobile.substring(0, 1));
 int length=mobile.length();
   if(length==10){
       if(initial_digit==8|| initial_digit==9 || initial_digit==7){
   String duplicate_check="select mobile_no from contact_info where
mobile_no="+mobile+";";
try{
    Statement stmt=connect();
    ResultSet rs=stmt.executeQuery(duplicate_check);
    while(rs.next()){
 String mysql_mobno=(rs.getString("mobile_no")).toLowerCase().trim();
      if(mobile.equals(mysql_mobno)){
            JOptionPane.showMessageDialog(null, "Sorry! this number has
already been taken by our existing user. please try a different mobile
number.");
            mobnoTf.setText("");
      }
   }
}
catch (Exception e){
           JOptionPane.showMessageDialog(null, e.getMessage());
      }
}
else{
          JOptionPane.showMessageDialog(null, "This kind of mobile number
does not exist in India. Please enter a VALID number.");
          mobnoTf.setText("");
    }
  }
else{
 JOptionPane.showMessageDialog(null, "Plz enter a 10 digit number");
      mobnoTf.setText("");
    }
  }
}
```

private void loginActionPerformed(java.awt.event.ActionEvent evt) {

```
new LOGIN_FORM().setVisible(true);
congratsMsg.setVisible(false);
}
```

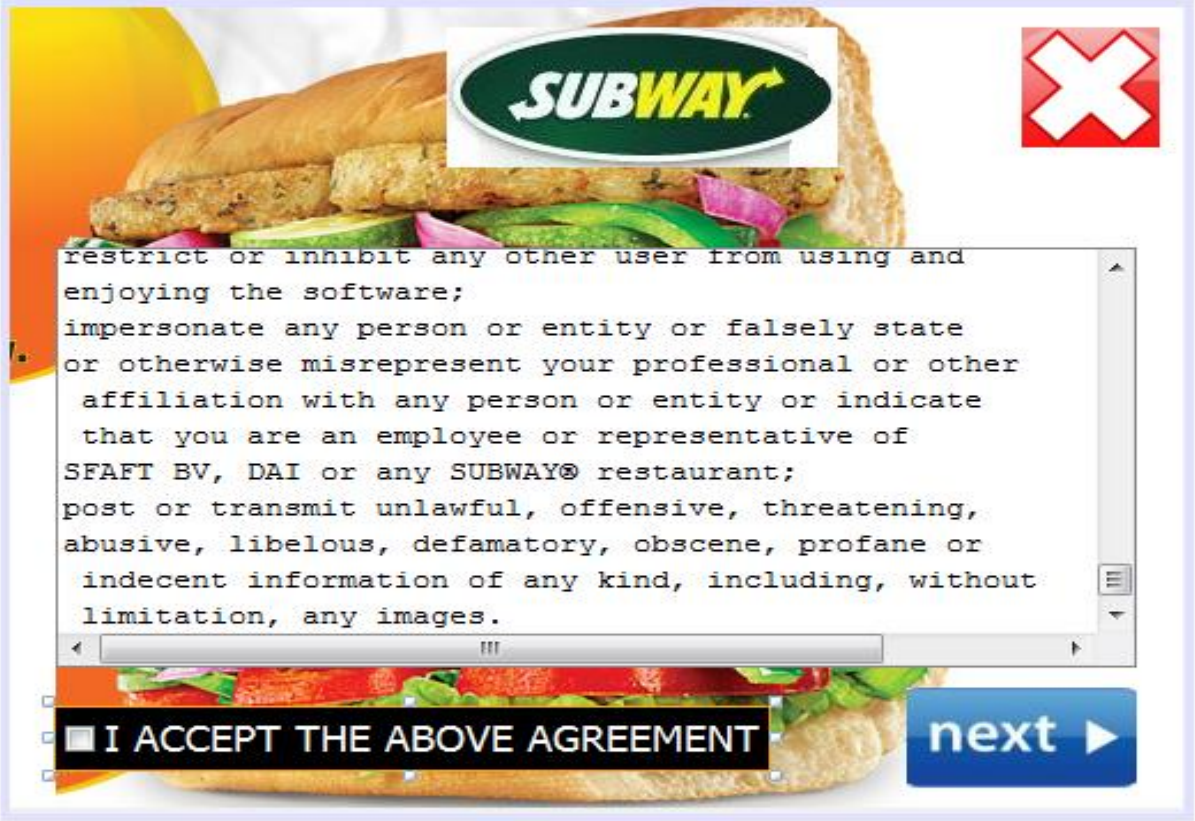


USERNAME 106

PASSWORD ronaldo

```
  private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
  new LOGIN_FORM().setVisible(true);
    dispose(); }
   public Statement connect() throws Exception{
     Class.forName("java.sql.Driver");
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/subway","root"
,"123");
Statement stmt = con.createStatement();
return stmt;
}
```

Main Frame: TERMS AND CONDITIONS



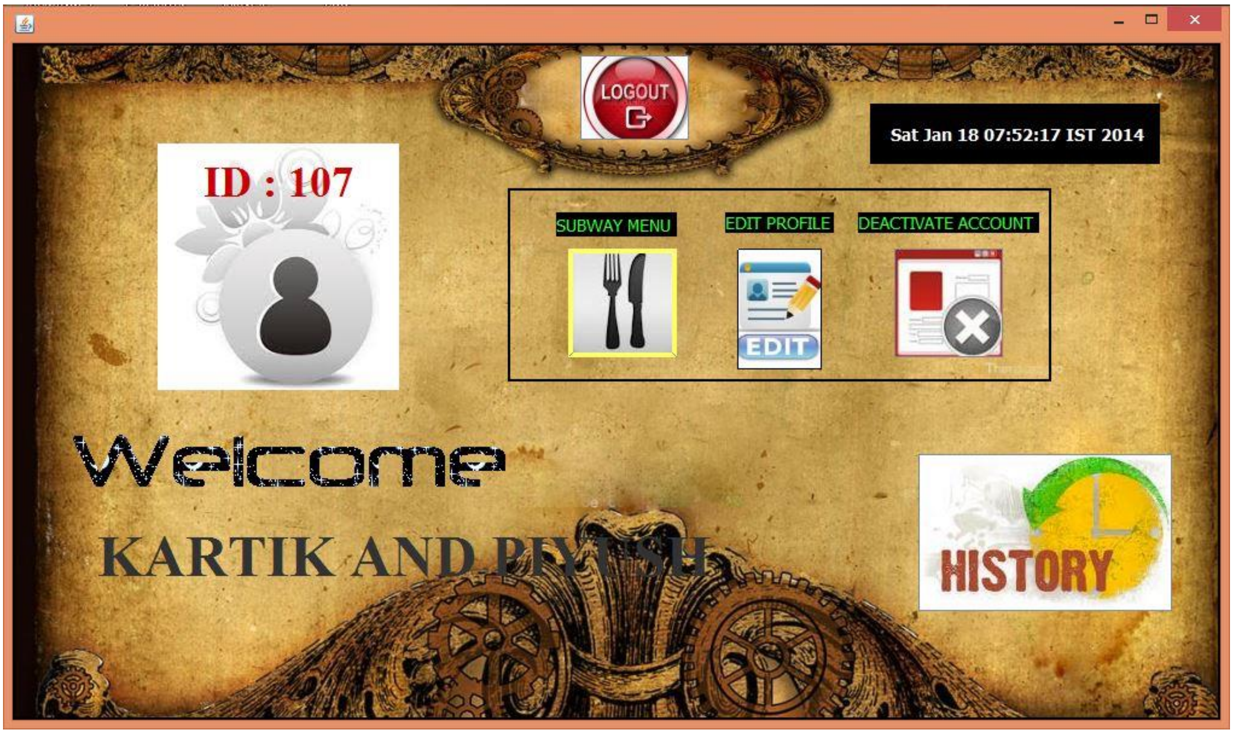
restrict or inhibit any other user from using and enjoying the software;
impersonate any person or entity or falsely state or otherwise misrepresent your professional or other affiliation with any person or entity or indicate that you are an employee or representative of SFAFT BV, DAI or any SUBWAY® restaurant;
post or transmit unlawful, offensive, threatening, abusive, libelous, defamatory, obscene, profane or indecent information of any kind, including, without limitation, any images.

SOURCE CODE: Terms and Conditions.java

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{

      new LOGIN_FORM().setVisible(true);

      dispose();

}

   private void cbItemStateChanged(java.awt.event.ItemEvent evt)
{

 if(cb.isSelected()){
   next.setEnabled(true);
}
 else
     next.setEnabled(false);
}
private void formWindowGainedFocus(java.awt.event.WindowEvent evt) {
   next.setEnabled(false);
}
 private void nextActionPerformed(java.awt.event.ActionEvent evt) {
      new become_a_member().setVisible(true);
      dispose();
}
```

USERS SECTION

Main Frame: USER DASHBOARD



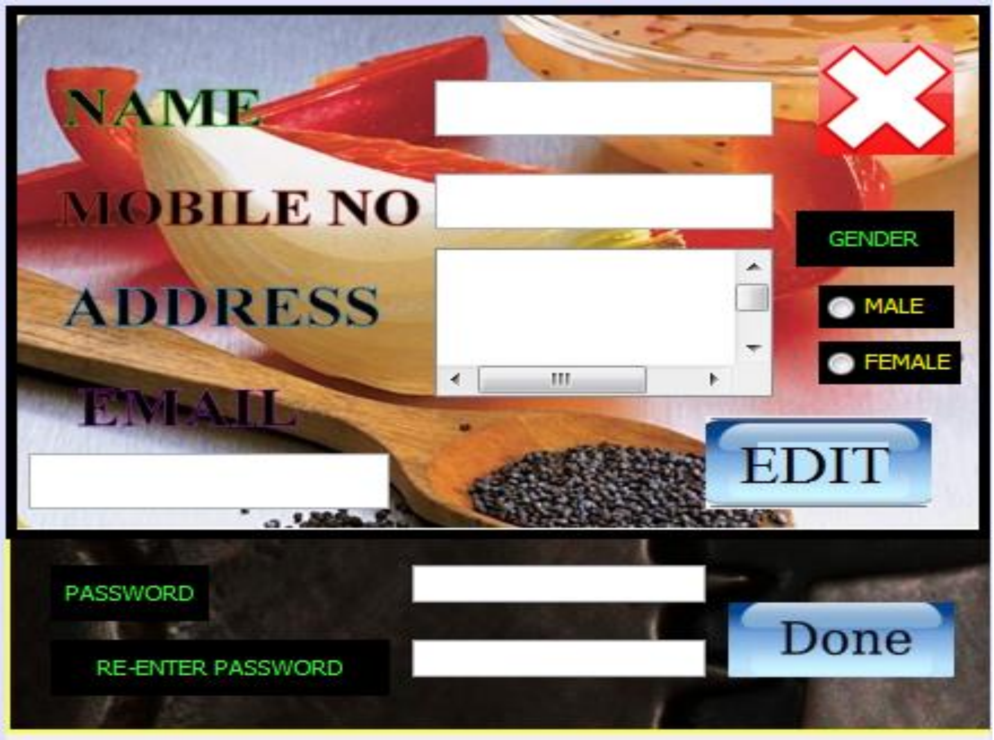
SOURCE CODE: User Dashboard.java

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{

String password= JOptionPane.showInputDialog(null, "Please
enter your password for security reasons.");

 try {

        Class.forName("java.sql.Driver");
```

```
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/
subway","root","123");

Statement stmt = con.createStatement();

String get_password = "select password from members
where status = 'online' and ID>101 ;";

ResultSet rs = stmt.executeQuery(get_password);

while(rs.next()){

    String actual_password=rs.getString("password");

 if(password.equals(actual_password)){

 int final_answer = JOptionPane.showConfirmDialog(null,
"ARE YOU SURE.. PROFILE ONCE DELETED CANNOT BE
RESTORED BACK");

 if(final_answer==0){

     String delete_records_from_members="delete from
members where status='online' ;";

     String delete_records_from_contact_info="delete from
contact_info where status='online' ;";

     String delete_records_from_orders="delete from orders
where logged_in='online' ;";


Statement stmt1=con.createStatement();
Statement stmt2=con.createStatement();

Statement stmt3=con.createStatement();

stmt1.executeUpdate(delete_records_from_members);

stmt2.executeUpdate(delete_records_from_contact_info);

stmt2.executeUpdate(delete_records_from_orders);


JOptionPane.showMessageDialog(null, "YOUR ACCOUNT
HAVE BEEN DELETED. THANK YOU FOR BEING WITH
SUBWAY.");

   new LOGIN_FORM().setVisible(true);

    dispose();

  }

}

  else{

    JOptionPane.showMessageDialog(null, "Incorrect password.
Please try again.");

  }

}

rs.close();

stmt.close();

con.close();

}
```

```
 catch(Exception e){

JOptionPane.showMessageDialog(this,e.getMessage());
   }
}
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{
          new main_menu().setVisible(true);
         dispose();
}
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{
      new edit_profile().setVisible(true);
       dispose();
}
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt)
{
        new LOGIN_FORM().setVisible(true);
           dispose();
}
private void formWindowGainedFocus(java.awt.event.WindowEvent evt) {
  Calendar c = Calendar.getInstance();
 datelb.setText(" " +c.getTime());
try{
          Class.forName("java.sql.Driver");
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/subway","
root","123");
Statement stmt = con.createStatement();
String get_user_info = "select name, ID  from members where
status='online' ;";
ResultSet rs=stmt.executeQuery(get_user_info);
 while(rs.next()){
String username_mysql=(rs.getString("name").toUpperCase());
  String id_mysql=rs.getString("ID");
  username.setText(""+username_mysql);
  id.setText("ID : "+id_mysql);
  }
}
catch(Exception e){
        JOptionPane.showMessageDialog(this,e.getMessage());
  }
}
```

Main Frame: User Profile



SOURCE CODE: User profile.java

```
private void gActionPerformed(java.awt.event.ActionEvent evt)
{

String name=a.getText().trim().toLowerCase();

String gender="";

 if(e.isSelected())

     gender = "M";

else

      gender = "F";
```

```
String address=c.getText().trim().toLowerCase();
String email_id = d.getText().trim().toLowerCase();
String password= "";
String password1 = new String(h.getPassword());
String password2 = new String(i.getPassword());
        if(password1.equals(password2)){
             password=password1;
     }
       else{
      JOptionPane.showMessageDialog(null, "the password you
entered didn’t matched, please try again");
            g.doClick();
    }
        if(name.equals("")||
password.equals("")||email_id.equals("")||address.equals(""))
JOptionPane.showMessageDialog(null, "All fields are mandatory.");
      else{
            try {
             long mobile = Long.parseLong(b.getText());
            Class.forName("java.sql.Driver");
```

```
Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:33
06/subway","root","123");

            Statement stmt=con.createStatement();

            Statement stmt1=con.createStatement();

            Statement stmt2=con.createStatement();

String update_members="update members set name='"+ name +"'
, gender='"+gender+"'  , password='"+password+"'where ID=101 ;";

String update_contact_info= "update contact_info  set name='"+
name +"'  , mobile_no="+mobile+" ,email='"+email_id+"'  ,
address='"+address+"' where ID=101 ;";

String update_orders= "update orders set name='"+ name +"'
where ID=101 ;";

        stmt.executeUpdate(update_members);

        stmt1.executeUpdate(update_contact_info);

        stmt2.executeUpdate(update_orders);

      stmt.close();

      con.close();

JOptionPane.showMessageDialog(null, "changes saved
successfully");

      new user_dashboard().setVisible(true);
```

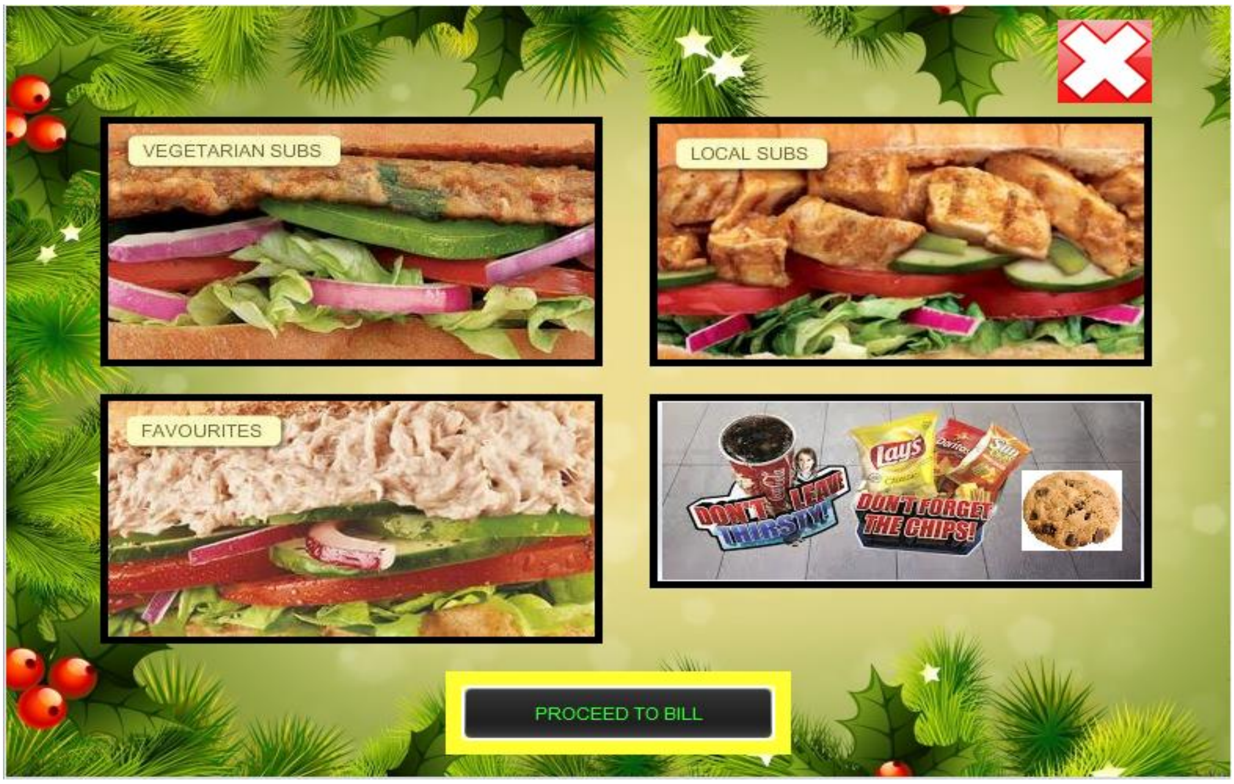
```
            dispose();
  }
    catch(Exception e){
JOptionPane.showMessageDialog(this,e.getMessage());
                      dispose();
    }
  }
}
    private void bFocusLost(java.awt.event.FocusEvent evt) {
        String mobile=b.getText().trim();
        if(mobile.equals("")){
  JOptionPane.showMessageDialog(null, "this field is mandatory.");
 }
   else{
    int initial_digit=Integer.parseInt(mobile.substring(0, 1));
       int length=mobile.length();
         if(length==10){
  if(initial_digit==8|| initial_digit==9 || initial_digit==7){
String duplicate_check="select mobile_no from contact_info
where mobile_no="+mobile+" and status='offline' and ID>101 ;";
```

```
 try{

          Class.forName("java.sql.Driver");

Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/
subway","root","123");

     Statement stmt=con.createStatement();

     ResultSet rs=stmt.executeQuery(duplicate_check);

    while(rs.next()){

String
mysql_mobno=(rs.getString("mobile_no")).toLowerCase().trim();

    if(mobile.equals(mysql_mobno)){

JOptionPane.showMessageDialog(null, "Sorry! this number
has already been taken by our existing user. please try a
different mobile number.");

                    b.setText("");

          }

       }

catch(Exception e){

      JOptionPane.showMessageDialog(null, e.getMessage());

   }

}
```

```
 else{
      JOptionPane.showMessageDialog(null, "This kind of
mobile number does not exist in India. Please enter a VALID
number.");
          b.setText("");
   }
 }
else{
JOptionPane.showMessageDialog(null, "Please enter a 10 digit
number ONLY");
              b.setText("");
     }
   }
}


private void bKeyTyped(java.awt.event.KeyEvent evt) {
       char ch = evt.getKeyChar();
       if( ((ch>='0') && (ch<='9')) ){
          evt.consume();
       }
```

```
    }
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{
        new owners_dashboard().setVisible(true);

        dispose();

                                                 }
```

Main Frame: Main Menu

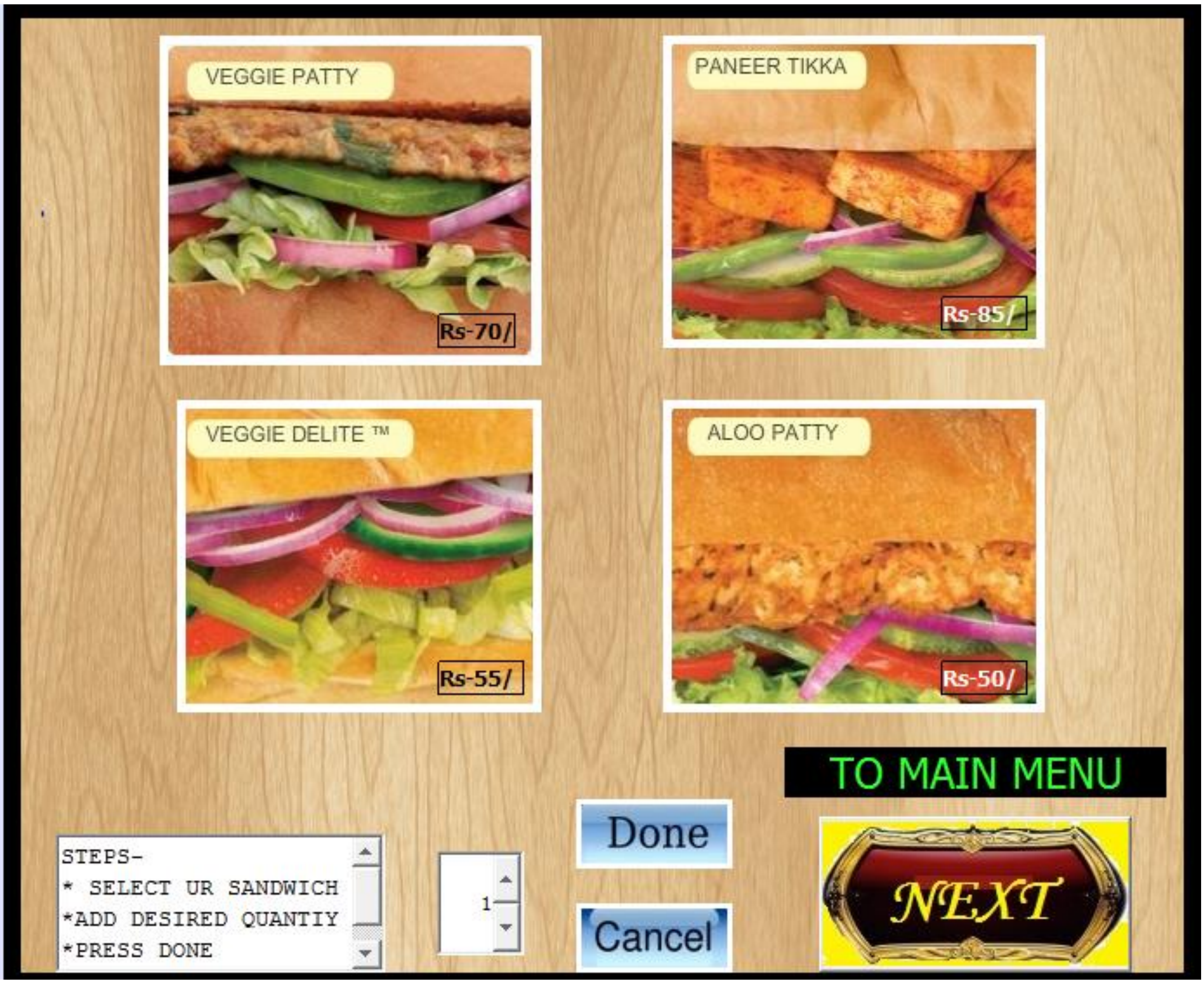


SOURCE CODE: Main Menu.java

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt){
  new menu_favourites().setVisible(true);
              dispose();
 }
private void jButton2ActionPerformed(java.awt.event.ActionEvent
evt){
   new Extras().setVisible(true);
          dispose();
```

```
    }
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
     new menu_local_subs().setVisible(true);
             dispose();
   }
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {
  new menu_vegetarian_subs().setVisible(true);
  dispose();
    }
private void formWindowGainedFocus(java.awt.event.WindowEvent evt)
{
try {
            Class.forName("java.sql.Driver");
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/
subway","root","123");
String query = "select count(name) as count from orders
where logged_in='online' and status='unpaid';";
Statement stmt=con.createStatement();
 ResultSet rs=stmt.executeQuery(query);
 while(rs.next()){
      int count=Integer.parseInt(rs.getString("count"));
          if(count==0){
           nextBtn.setText("NO BILLS PENDING");
               nextBtn.setEnabled(false);
  }
     else{
           nextBtn.setText("PROCEED TO BILL");
      }
   }
 }
 catch (Exception e){
       JOptionPane.showMessageDialog(null, e.getMessage());
   }
}
private void nextBtnActionPerformed(java.awt.event.ActionEvent evt) {
new bill().setVisible(true);
   dispose();
}
private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
       new user_dashboard().setVisible(true); dispose(); }
```

Main Frame: VEGETERIAN SUB



```
STEPS-
* SELECT UR SANDWICH
*ADD DESIRED QUANTITY
*PRESS DONE
```

SOURCE CODE: Vegeterian sub.java

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{

    doneBtn.setEnabled(true);

    cancelBtn.setEnabled(true);
```

```
     jButton2.setEnabled(false);

    jButton3.setEnabled(false);

    jButton4.setEnabled(false);

}
```

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{

```
    doneBtn.setEnabled(true);

    cancelBtn.setEnabled(true);

    jButton1.setEnabled(false);

    jButton3.setEnabled(false);

    jButton4.setEnabled(false);

}
```

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{

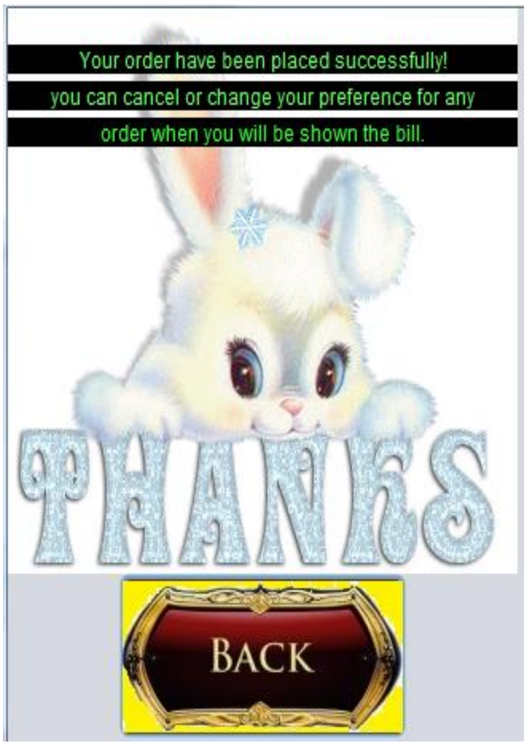
```
     doneBtn.setEnabled(true);

     cancelBtn.setEnabled(true);

     jButton1.setEnabled(false);

     jButton2.setEnabled(false);

     jButton4.setEnabled(false);

}
```

```
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt)
{
      doneBtn.setEnabled(true);
      cancelBtn.setEnabled(true);
      jButton1.setEnabled(false);;
      jButton2.setEnabled(false);
      jButton3.setEnabled(false);
}
private void cancelBtnActionPerformed(java.awt.event.ActionEvent evt) {
     jButton1.setEnabled(true);
     jButton2.setEnabled(true);
     jButton3.setEnabled(true);
     jButton4.setEnabled(true);
     doneBtn.setEnabled(false);
     cancelBtn.setEnabled(false);
}
private void doneBtnActionPerformed(java.awt.event.ActionEvent evt)
{
try {
         Class.forName("java.sql.Driver");
```

```
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/
subway","root","123");
String query="select ID from members where status='online';";
String query2="select name from members where status='online';";
String sno_query="select sno from orders order by sno desc;";
    Statement stmt = con.createStatement();
    ResultSet rs = stmt.executeQuery(query);
  while(rs.next()){
  int ID_mysql=Integer.parseInt(rs.getString("ID"));

  Statement stmt1 = con.createStatement();
  rs = stmt1.executeQuery(query2);
  while(rs.next()){
  String name_mysql=rs.getString("name");
  Statement stmt3=con.createStatement();
  rs = stmt3.executeQuery(sno_query);
  while(rs.next()){
  int sno = Integer.parseInt(rs.getString("sno"));
    String item = " ";
     int price;
```

```
      if(jButton1.isEnabled()){
       item="Vegie patty Sandwich";
      price=70;
      priceLb.setText(""+price);
}
  else
        if(jButton2.isEnabled()){
        item="CVegie delite Sandwich";
        price=55;
        priceLb.setText(""+price);
 }
   else
         if(jButton3.isEnabled()){
         item="Paneer Tikka Sandwich";
         price=80;
         priceLb.setText(""+price);
 }
    else
          if(jButton4.isEnabled()){
         item="Aloo Patty Sandwich";
        price=50;

        priceLb.setText(""+price);

}

      Calendar c = Calendar.getInstance();

   int price_final= Integer.parseInt(priceLb.getText());

String order="insert into orders
values("+(sno+1)+","+ID_mysql+",'online','"+name_mysql+"','"+item+"',"
+Integer.parseInt(priceLb.getText())+","+quantity+",'"+(c.get(Calenda
r.YEAR))+"-"+(c.get(Calendar.MONTH)+1)+"-
"+(c.get(Calendar.DATE))+"','unpaid');" ;

  Statement stmt4=con.createStatement();

  stmt4.executeUpdate(order);

String total_collection= "update total_collection set
total_money = ( total_money + " + price_final*quantity + ") ,
total_orders = total_orders +1  ;";

String new_order_placed_today ="insert into total_collection_perday
values ('order placed','"+(c.get(Calendar.YEAR))+"-
"+(c.get(Calendar.MONTH)+1)+"-
"+(c.get(Calendar.DATE))+"',"+price_final*quantity+");";

   Statement stmt5=con.createStatement();

   Statement stmt6=con.createStatement();

   stmt5.executeUpdate(total_collection);

    stmt6.executeUpdate(new_order_placed_today);
   }
  }
}
    jButton1.setEnabled(true);
    jButton2.setEnabled(true);
    jButton3.setEnabled(true);
    jButton4.setEnabled(true);
    cancelBtn.setEnabled(false);
   doneBtn.setEnabled(false);
   thanksMsg.setVisible(true);
    dispose();
catch(Exception e){
   jButton1.setEnabled(true);
   jButton2.setEnabled(true);
   jButton3.setEnabled(true);
  jButton4.setEnabled(true);
  cancelBtn.setEnabled(false);
  doneBtn.setEnabled(false);
   thanksMsg.setVisible(true);
 dispose();
```

```
  }

}

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt)
{

        thanksMsg.setVisible(false);

       new menu_favourites().setVisible(true);

}
```

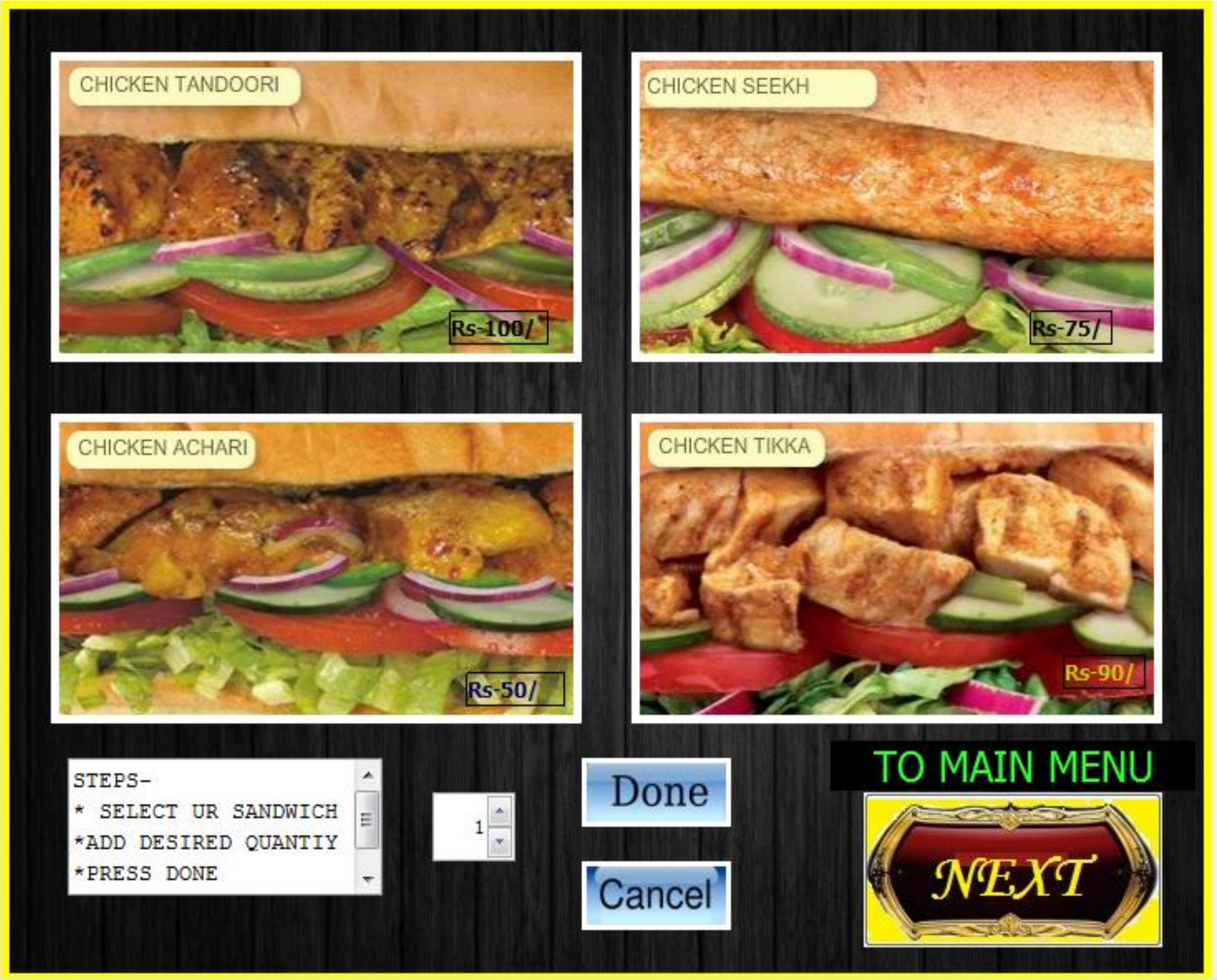


Your order have been placed successfully!

you can cancel or change your preference for any

order when you will be shown the bill.

Main Frame: LOCAL SUBS

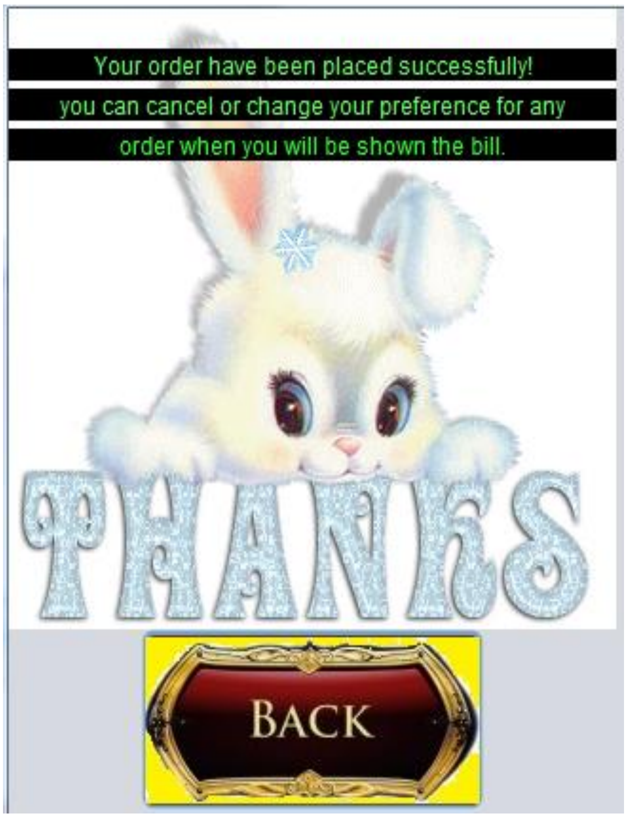


SOURCE CODE: Local subs.java

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
doneBtn.setEnabled(true);
cancelBtn.setEnabled(true);
jButton2.setEnabled(false);
jButton3.setEnabled(false);
```

```
  jButton4.setEnabled(false);

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{

   doneBtn.setEnabled(true);

   cancelBtn.setEnabled(true);

   jButton1.setEnabled(false);

   jButton3.setEnabled(false);

   jButton4.setEnabled(false);

}
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{

    doneBtn.setEnabled(true);

    cancelBtn.setEnabled(true);

    jButton1.setEnabled(false);

    jButton2.setEnabled(false);

    jButton4.setEnabled(false);

}
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt)
{

    doneBtn.setEnabled(true);
    cancelBtn.setEnabled(true);

    jButton1.setEnabled(false);;

    jButton2.setEnabled(false);

    jButton3.setEnabled(false);

}
private void cancelBtnActionPerformed(java.awt.event.ActionEvent evt) {
     jButton1.setEnabled(true);

     jButton2.setEnabled(true);

     jButton3.setEnabled(true);

     jButton4.setEnabled(true);

     doneBtn.setEnabled(false);

     cancelBtn.setEnabled(false);

 }
private void doneBtnActionPerformed(java.awt.event.ActionEvent evt)
{
try {
          Class.forName("java.sql.Driver");
Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/
subway","root","123");
String query = "select ID from members where status ='online';";
String query2 = "select name from members where status='online';";

String sno_query="select sno from orders order by sno desc;";

    Statement stmt=con.createStatement();

    ResultSet rs=stmt.executeQuery(query);

while(rs.next()){

  int ID_mysql=Integer.parseInt(rs.getString("ID"));


  Statement stmt1 = con.createStatement();

  rs = stmt1.executeQuery(query2);

  while(rs.next()){

  String name_mysql=rs.getString("name");


  Statement stmt3=con.createStatement();

  rs=stmt3.executeQuery(sno_query);

 while(rs.next()){

  int sno=Integer.parseInt(rs.getString("sno"));

String item=" ";

      int price;

if(jButton1.isEnabled()){

      item=" Chicken Tandoori sandwich ";

      price=100;

      priceLb.setText(""+price);
}
 else
      if(jButton2.isEnabled()){
     item=" Chicken achari sandwich ";
     price=50;
     priceLb.setText(""+price);
 }
 else
      if(jButton3.isEnabled()){
      item="Chicken seekh sandwich";
      price=75;
     priceLb.setText(""+price);
 }
  else
      if(jButton4.isEnabled()){
      item=" Chicken tikka sandwich";
      price=90;
      priceLb.setText(""+price);
}
  Calendar c = Calendar.getInstance();
 int price_final = Integer.parseInt(priceLb.getText());
```

```
String order = "insert into orders
values("+(sno+1)+","+ID_mysql+",'online','"+name_mysql+"','"+item+"',"
+Integer.parseInt(priceLb.getText())+","+quantity+",'"+(c.get(Calenda
r.YEAR))+"-"+(c.get(Calendar.MONTH)+1)+"-
"+(c.get(Calendar.DATE))+"','unpaid');" ;

   Statement stmt4 = con.createStatement();

   stmt4.executeUpdate(order);

 String total_collection= "update total_collection set total_money = (
total_money + " + price_final*quantity + ") , total_orders =
total_orders +1  ;";

String new_order_placed_today ="insert into total_collection_perday
values ('order placed','"+(c.get(Calendar.YEAR))+"-
"+(c.get(Calendar.MONTH)+1)+"-
"+(c.get(Calendar.DATE))+"',"+price_final*quantity+");";

    Statement stmt5=con.createStatement();

    Statement stmt6=con.createStatement();

    stmt5.executeUpdate(total_collection);

    stmt6.executeUpdate(new_order_placed_today);

   }

  }

}

catch(Exception e){

   jButton1.setEnabled(true);

   jButton2.setEnabled(true);
   jButton3.setEnabled(true);

   jButton4.setEnabled(true);

   cancelButton.setEnabled(false);

   doneButton.setEnabled(false);

   thanksMsg.setVisible(true);

    dispose();

 }

}
```



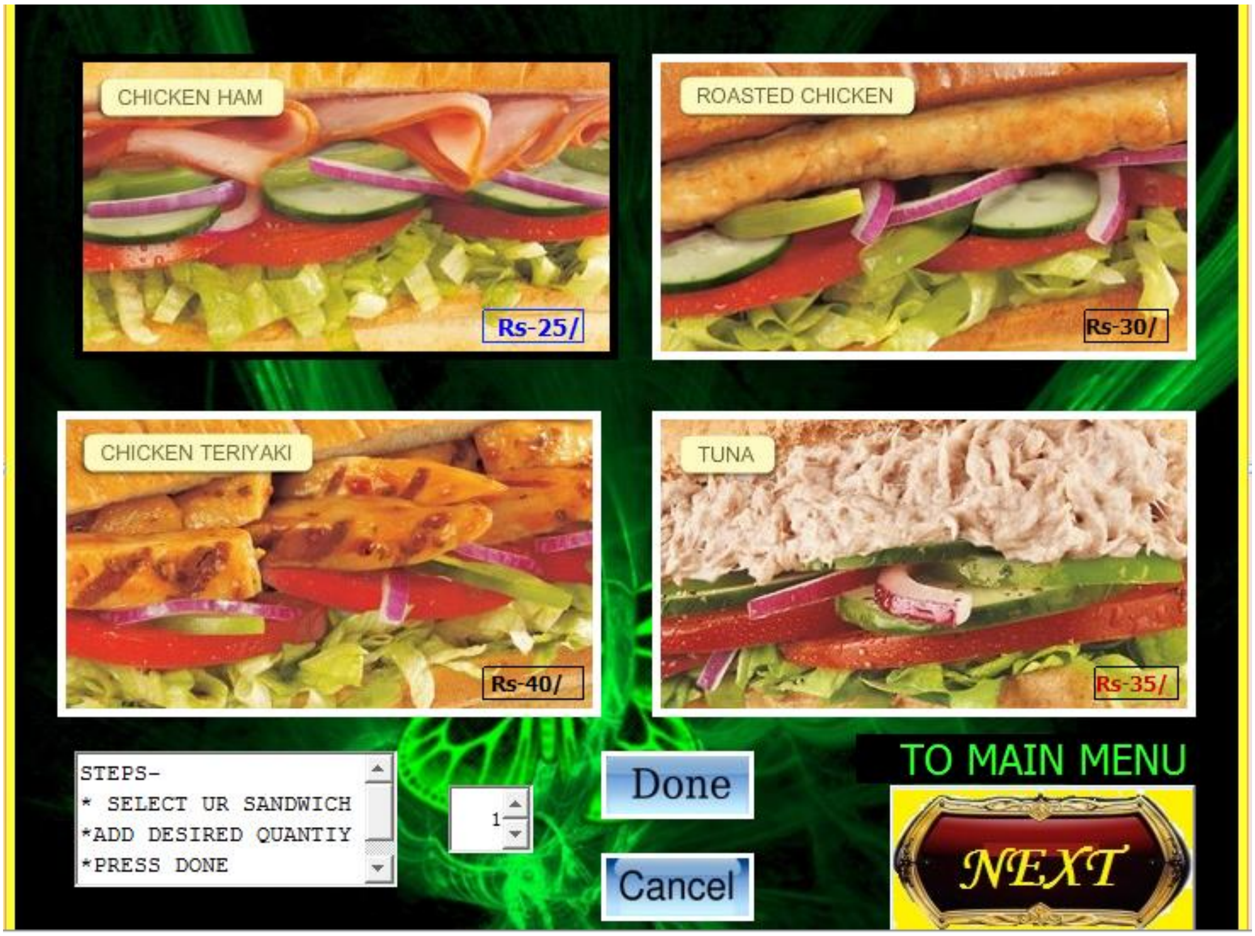
Your order have been placed successfully!

you can cancel or change your preference for any

order when you will be shown the bill.

```
private void jButton6ActionPerformed(java.awt.event.ActionEvent evt)
{

        thanksMsg.setVisible(false);

       new menu_favourites().setVisible(true);  }
```

Main Frame: FAVOURITES



```
STEPS-
* SELECT UR SANDWICH
*ADD DESIRED QUANTIY
*PRESS DONE
```

SOURCE CODE: Favourites.java

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
  doneBtn.setEnabled(true);
  cancelBtn.setEnabled(true);
  jButton2.setEnabled(false);
  jButton3.setEnabled(false);
  jButton4.setEnabled(false);
```

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{

    doneBtn.setEnabled(true);

  cancelBtn.setEnabled(true);

  jButton1.setEnabled(false);

  jButton3.setEnabled(false);

  jButton4.setEnabled(false);

}
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{

   doneBtn.setEnabled(true);

   cancelBtn.setEnabled(true);

   jButton1.setEnabled(false);

   jButton2.setEnabled(false);

   jButton4.setEnabled(false);

}
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt)
{

   doneBtn.setEnabled(true);

   cancelBtn.setEnabled(true);

    jButton1.setEnabled(false);;
   jButton2.setEnabled(false);

   jButton3.setEnabled(false);

}
```

private void cancelBtnActionPerformed(java.awt.event.ActionEvent evt) {

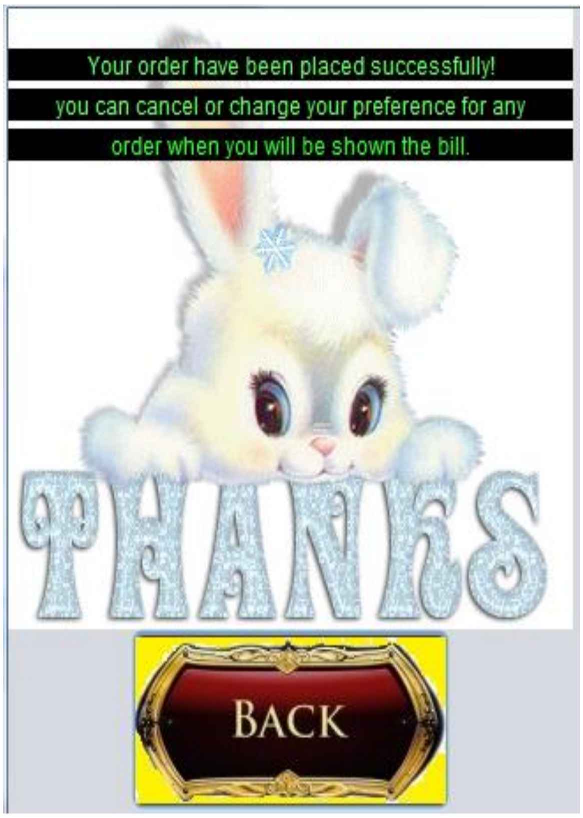
```
    jButton1.setEnabled(true);

   jButton2.setEnabled(true);

   jButton3.setEnabled(true);

   jButton4.setEnabled(true);

   doneBtn.setEnabled(false);

   cancelBtn.setEnabled(false);

 }
```

private void doneBtnActionPerformed(java.awt.event.ActionEvent evt) {

```
try {

        Class.forName("java.sql.Driver");

Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/
subway","root","123");

String query = "select ID from members where status ='online';";

String query2 = "select name from members where status='online';";

String sno_query="select sno from orders order by sno desc;";
```

```
    Statement stmt=con.createStatement();

    ResultSet rs=stmt.executeQuery(query);

while(rs.next()){

  int ID_mysql=Integer.parseInt(rs.getString("ID"));

  Statement stmt1 = con.createStatement();

  rs = stmt1.executeQuery(query2);

  while(rs.next()){

  String name_mysql=rs.getString("name");

  Statement stmt3=con.createStatement();

  rs=stmt3.executeQuery(sno_query);

 while(rs.next()){

  int sno=Integer.parseInt(rs.getString("sno"));

String item=" ";

      int price;

if(jButton1.isEnabled()){

      item="Chicken Ham Sandwich";

      price=25;

      priceLb.setText(""+price);

 }
 else
      if(jButton2.isEnabled()){
     item="Chicken Teriyaki Sandwich";
     price=40;
     priceLb.setText(""+price);
 }
 else
       if(jButton3.isEnabled()){
       item="Roasted Chicken sandwich";
       price=30;
      priceLb.setText(""+price);
 }
  else
       if(jButton4.isEnabled()){
       item="Tuna sandwich";
       price=35;
       priceLb.setText(""+price);
}
  Calendar c = Calendar.getInstance();
 int price_final = Integer.parseInt(priceLb.getText());
```

```
String order = "insert into orders
values("+(sno+1)+","+ID_mysql+",'online','"+name_mysql+"','"+item
+"',"+Integer.parseInt(priceLb.getText())+","+quantity+",'"+(c.get
(Calendar.YEAR))+"-"+(c.get(Calendar.MONTH)+1)+"-
"+(c.get(Calendar.DATE))+"','unpaid');" ;

   Statement stmt4 = con.createStatement();

   stmt4.executeUpdate(order);

 String total_collection= "update total_collection set total_money = (
total_money + " + price_final*quantity + ") , total_orders =
total_orders +1  ;";

String new_order_placed_today ="insert into total_collection_perday
values ('order placed','"+(c.get(Calendar.YEAR))+"-
"+(c.get(Calendar.MONTH)+1)+"-
"+(c.get(Calendar.DATE))+"',"+price_final*quantity+");";

    Statement stmt5=con.createStatement();

    Statement stmt6=con.createStatement();

    stmt5.executeUpdate(total_collection);

    stmt6.executeUpdate(new_order_placed_today);

   }

  }

}

catch(Exception e){

   jButton1.setEnabled(true);

   jButton2.setEnabled(true);
   jButton3.setEnabled(true);

   jButton4.setEnabled(true);

   cancelButton.setEnabled(false);

   doneButton.setEnabled(false);

   thanksMsg.setVisible(true);

    dispose();

 }

}
```



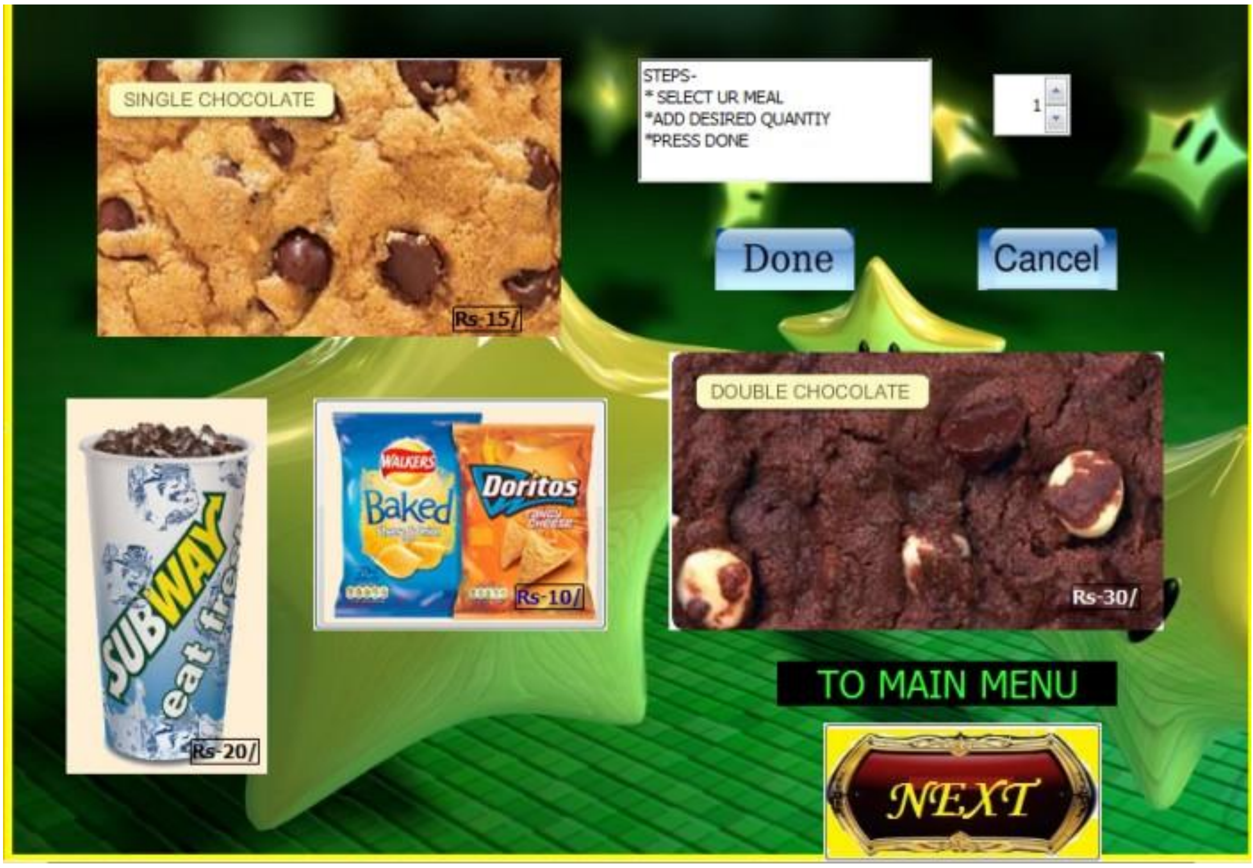
Your order have been placed successfully!

you can cancel or change your preference for any

order when you will be shown the bill.

```
private void jButton6ActionPerformed(java.awt.event.ActionEvent evt)
{

         thanksMsg.setVisible(false);

        new menu_favourites().setVisible(true); }
```

Main Frame: EXTRAS



STEPS-
* SELECT UR MEAL
*ADD DESIRED QUANTITY
*PRESS DONE

SOURCE CODE: Extras.java

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
  doneBtn.setEnabled(true);

  cancelBtn.setEnabled(true);

  jButton2.setEnabled(false);

  jButton3.setEnabled(false);

  jButton4.setEnabled(false);

}
```

```
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{
   doneBtn.setEnabled(true);
   cancelBtn.setEnabled(true);
   jButton1.setEnabled(false);
   jButton3.setEnabled(false);
   jButton4.setEnabled(false);
}
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{
    doneBtn.setEnabled(true);
    cancelBtn.setEnabled(true);
    jButton1.setEnabled(false);
    jButton2.setEnabled(false);
    jButton4.setEnabled(false);
}
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt)
{
    doneBtn.setEnabled(true);
    cancelBtn.setEnabled(true);
    jButton1.setEnabled(false);;
    jButton2.setEnabled(false);

    jButton3.setEnabled(false);

}
```

private void cancelBtnActionPerformed(java.awt.event.ActionEvent evt) {

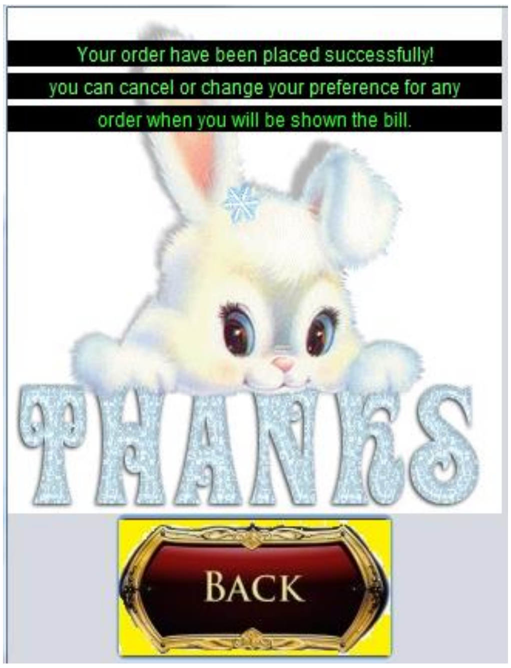
```
     jButton1.setEnabled(true);

     jButton2.setEnabled(true);

     jButton3.setEnabled(true);

     jButton4.setEnabled(true);

     doneBtn.setEnabled(false);

     cancelBtn.setEnabled(false);

 }
```

private void doneBtnActionPerformed(java.awt.event.ActionEvent evt) {

```
try {

           Class.forName("java.sql.Driver");

Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/subway","root","123");

String query="select ID from members where status='online';";

String query2="select name from members where status='online';";

 String sno_query="select sno from orders order by sno desc;";
```

```
   Statement stmt=con.createStatement();
   ResultSet rs=stmt.executeQuery(query);
  while(rs.next()){
 int ID_mysql=Integer.parseInt(rs.getString("ID"));

 Statement stmt1 = con.createStatement();
 rs = stmt1.executeQuery(query2);
 while(rs.next()){
 String name_mysql=rs.getString("name");

 Statement stmt3=con.createStatement();
 rs = stmt3.executeQuery(sno_query);
 while(rs.next()){
 int sno = Integer.parseInt(rs.getString("sno"));
  String item="";
   int price;
    if(jButton1.isEnabled()){
     item="Double chocolate cookie";
     price=30;
     priceLb.setText(""+price);
}
```

```
   else

        if(jButton2.isEnabled()){

        item="Chips";

        price=10;

        priceLb.setText(""+price);

}

   else

         if(jButton3.isEnabled()){

         item="Chocolate Cookie";

         price=15;

         priceLb.setText(""+price);

 }

   else

         if(jButton4.isEnabled()){

        item="Coke";

        price=20;

        priceLb.setText(""+price);

 }

   Calendar c = Calendar.getInstance();

  int price_final = Integer.parseInt(priceLb.getText());
 String order ="insert into orders
values("+(sno+1)+","+ID_mysql+",'online','"+name_mysql+"','"+
item+"',"+Integer.parseInt(priceLb.getText())+","+quantity+",
'"+(c.get(Calendar.YEAR))+"-"+(c.get(Calendar.MONTH)+1)+"-
"+(c.get(Calendar.DATE))+"','unpaid');" ;

    Statement stmt4=con.createStatement();

    stmt4.executeUpdate(order);

 String total_collection= "update total_collection set
total_money = ( total_money + " + price_final*quantity + ") ,
total_orders = total_orders +1  ;";

String new_order_placed_today ="insert into total_collection_perday
values ('order placed','"+(c.get(Calendar.YEAR))+"-
"+(c.get(Calendar.MONTH)+1)+"-
"+(c.get(Calendar.DATE))+"',"+price_final*quantity+");";

   Statement stmt5 = con.createStatement();

    Statement stmt6=con.createStatement();

    stmt5.executeUpdate(total_collection);

    stmt6.executeUpdate(new_order_placed_today);

   }

  }

}

     jButton1.setEnabled(true);
     jButton2.setEnabled(true);

     jButton3.setEnabled(true);

     jButton4.setEnabled(true);

     cancelBtn.setEnabled(false);

     doneBtn.setEnabled(false);

     thanksMsg.setVisible(true);

     dispose();

}

catch(Exception e){

      jButton1.setEnabled(true);

      jButton2.setEnabled(true);

    jButton3.setEnabled(true);

   jButton4.setEnabled(true);

   cancelBtn.setEnabled(false);

   doneBtn.setEnabled(false);

   thanksMsg.setVisible(true);

    dispose();

  }

}
```



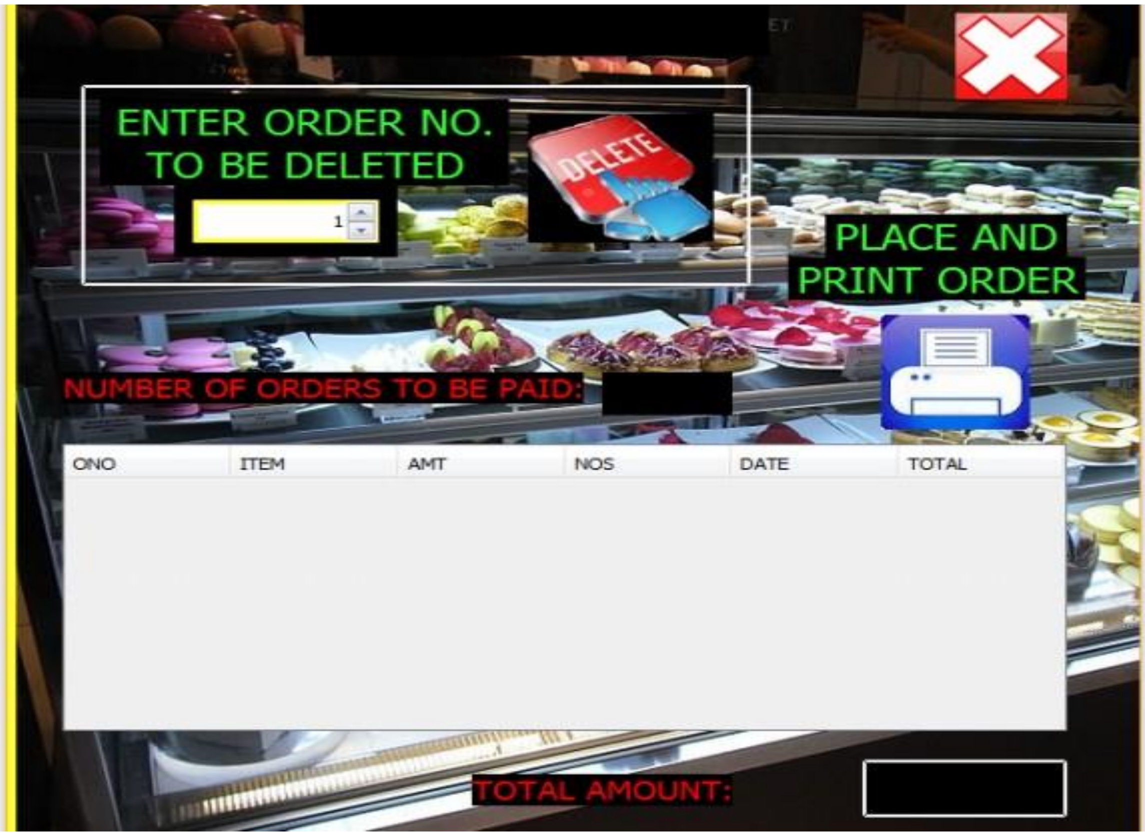
Your order have been placed successfully!

you can cancel or change your preference for any

order when you will be shown the bill.

```
private void jButton6ActionPerformed(java.awt.event.ActionEvent evt)
{

        thanksMsg.setVisible(false);

      new menu_favourites().setVisible(true);

}
```

Main Frame: Bill



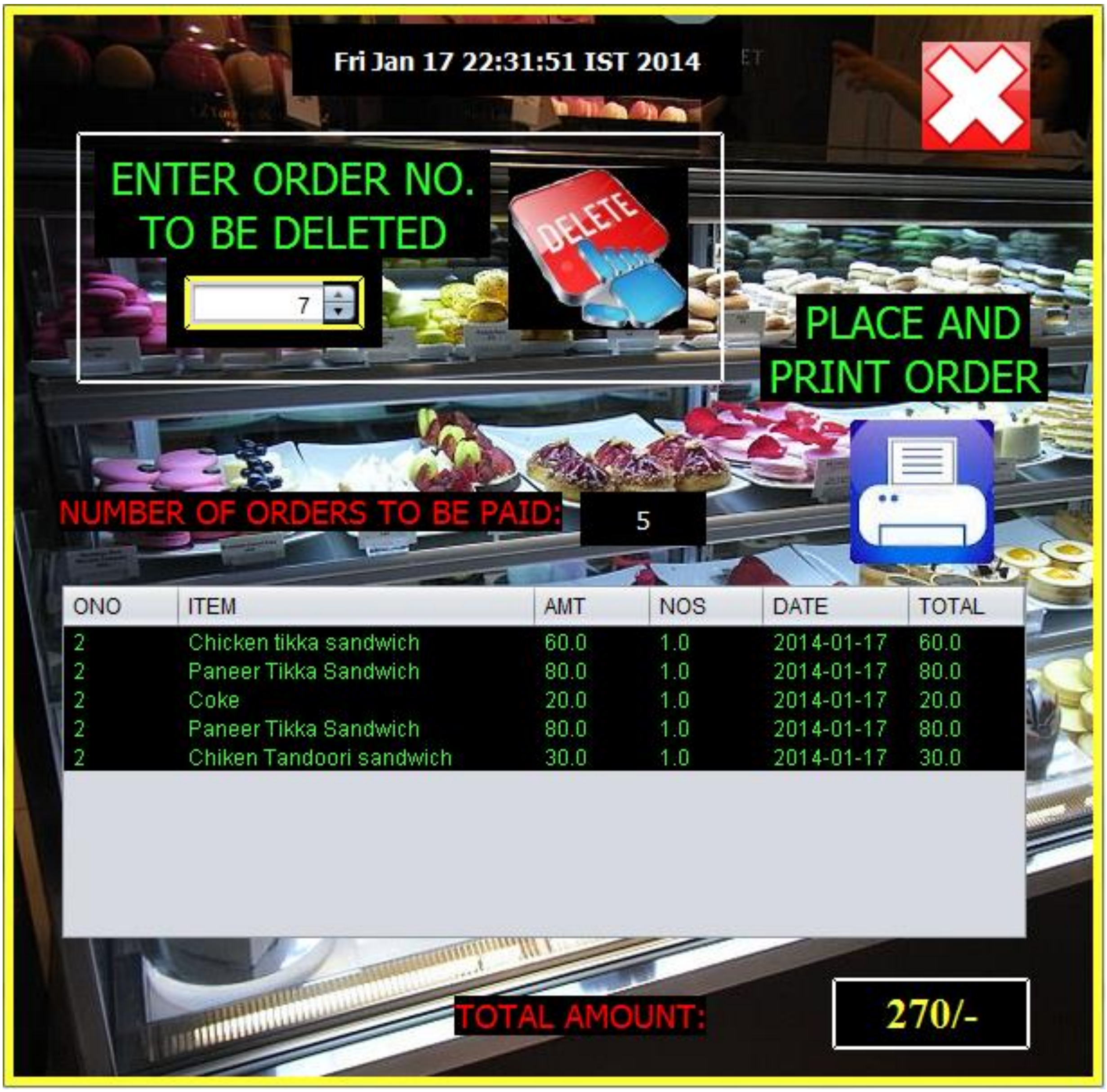
SOURCE CODE: Bill.java

```
public void add_records_in_table(){
   try{
         Class.forName("java.sql.Driver");
Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/subway","root","123");
Statement stmt=con.createStatement();
Statement stmt1=con.createStatement();
```

```
String count_of_records = "select count(*) as count from
orders where logged_in='online' and status='unpaid';";

String orders="select * from orders where
logged_in='online' and status='unpaid';";

ResultSet rs = stmt1.executeQuery(count_of_records);
while(rs.next()){
   int number=Integer.parseInt(rs.getString("count"));
no.setText(""+ number);
DefaultTableModel tables1 =(DefaultTableModel)info.getModel() ;
       int count=info.getRowCount();
  int m;
    for(m=count; m>0; m--){
      tables1.removeRow(0);
   }
      info.setModel(tables1);
   rs = stmt.executeQuery(orders);
 int i;
 int sum=0;
for(i=number;i>=0;i--){
while(rs.next()){
int sno = Integer.parseInt(rs.getString("sno"));
String item_ordered = rs.getString("item_ordered");
double amount = Double.parseDouble(rs.getString("amount"));
double quantity=Double.parseDouble(rs.getString("quantity"));
String date_ordered=rs.getString("date_ordered");
 double total= (amount*quantity);
   DefaultTableModel
tables=(DefaultTableModel)info.getModel();
 Object [] names_found = {sno, item_ordered, amount,
quantity, date_ordered,total};
    tables.addRow(names_found);
    sum += total;
  }
total.setText(""+sum+"/-");
}
  }
}
catch(Exception e){
    JOptionPane.showMessageDialog(null, e.getMessage());
  }

}
   private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

    new main_menu().setVisible(true);

    dispose();

    }

 private void formWindowGainedFocus(java.awt.event.WindowEvent evt) {

 Calendar c=Calendar.getInstance();

date.setText("" +c.getTime());

add_records_in_table();

}
 private void jButton2ActionPerformed(java.awt.event.ActionEvent
evt) {

        int order_no=(Integer)(sp.getValue());

 try {

            Class.forName("java.sql.Driver");

Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/
subway","root","123");

String delete_records_from_orders="delete from orders
where sno ="+ order_no +";";
```

```
 Statement stmt1 = con.createStatement();

 stmt1.executeUpdate(delete_records_from_orders);

      refresh.doClick();

}

 catch(Exception e){

  JOptionPane.showMessageDialog(this,e.getMessage());

   }

 }

    private void refreshActionPerformed(java.awt.event.ActionEvent
evt) {

  add_records_in_table();

}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt)
{

int confirm=JOptionPane.showConfirmDialog(null, "ORDER
ONCE PLACED CANNOT BE CANCELLED. ARE YOU SURE.");

  if(confirm==0){

try {

       Class.forName("java.sql.Driver");

Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/subway","
root","123");
```

```
                Statement stmt=con.createStatement();
String update_orders= "update orders set status = 'paid' where
logged_in = 'online';";
                stmt.executeUpdate(update_orders);
              new user_dashboard().setVisible(true);
  JOptionPane.showMessageDialog(null, "YOUR ORDER HAVE
BEEN PLACED. PLEASE COLLECT YOUR MEAL AT THE COUNTER
AND PAY THE MONEY. THANKYOU. ")
                         dispose();
 }
catch(Exception e){
     JOptionPane.showMessageDialog(null,e.getMessage());
 }
```



Fri Jan 17 22:31:51 IST 2014

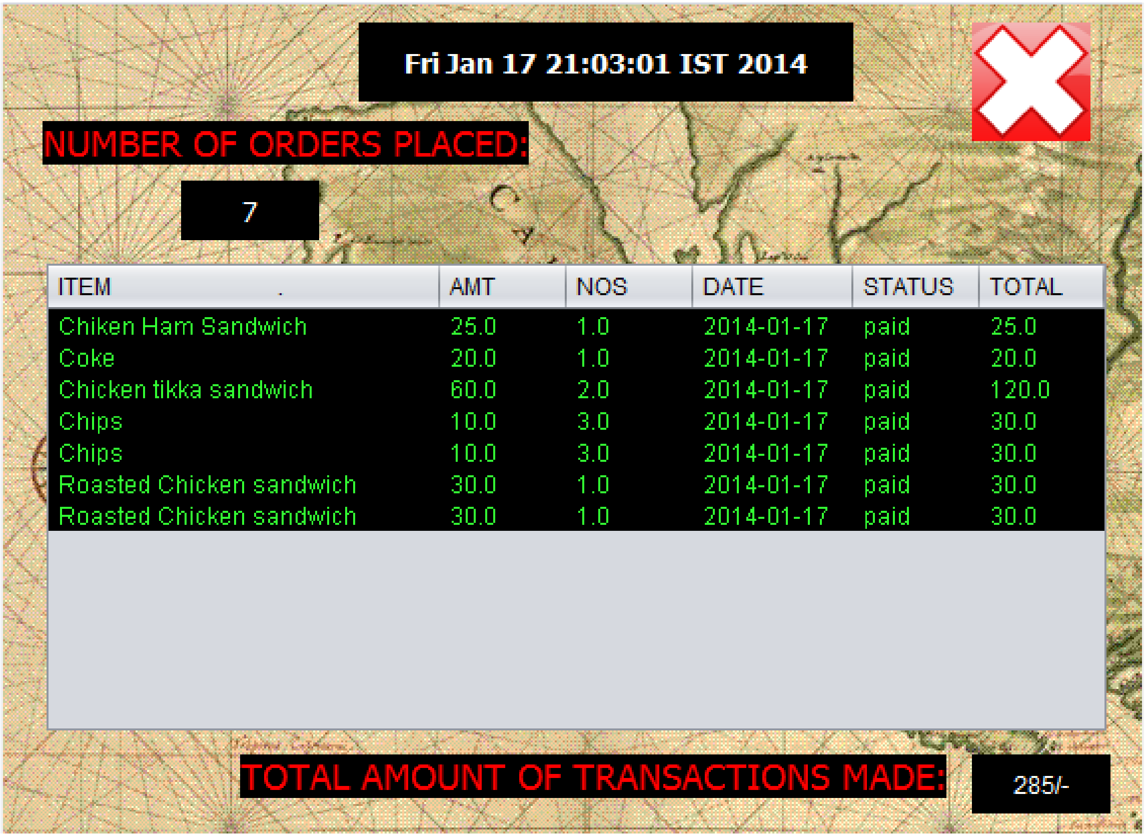
ENTER ORDER NO. TO BE DELETED: 7

NUMBER OF ORDERS TO BE PAID: 5

ONO	ITEM	AMT	NOS	DATE	TOTAL
2	Chicken tikka sandwich	60.0	1.0	2014-01-17	60.0
2	Paneer Tikka Sandwich	80.0	1.0	2014-01-17	80.0
2	Coke	20.0	1.0	2014-01-17	20.0
2	Paneer Tikka Sandwich	80.0	1.0	2014-01-17	80.0
2	Chiken Tandoori sandwich	30.0	1.0	2014-01-17	30.0

TOTAL AMOUNT: 270/-

Main Frame: HISTORY



Fri Jan 17 21:03:01 IST 2014

NUMBER OF ORDERS PLACED:

7

ITEM	AMT	NOS	DATE	STATUS	TOTAL
Chiken Ham Sandwich	25.0	1.0	2014-01-17	paid	25.0
Coke	20.0	1.0	2014-01-17	paid	20.0
Chicken tikka sandwich	60.0	2.0	2014-01-17	paid	120.0
Chips	10.0	3.0	2014-01-17	paid	30.0
Chips	10.0	3.0	2014-01-17	paid	30.0
Roasted Chicken sandwich	30.0	1.0	2014-01-17	paid	30.0
Roasted Chicken sandwich	30.0	1.0	2014-01-17	paid	30.0

TOTAL AMOUNT OF TRANSACTIONS MADE:

285/-

SOURCE CODE: History.java

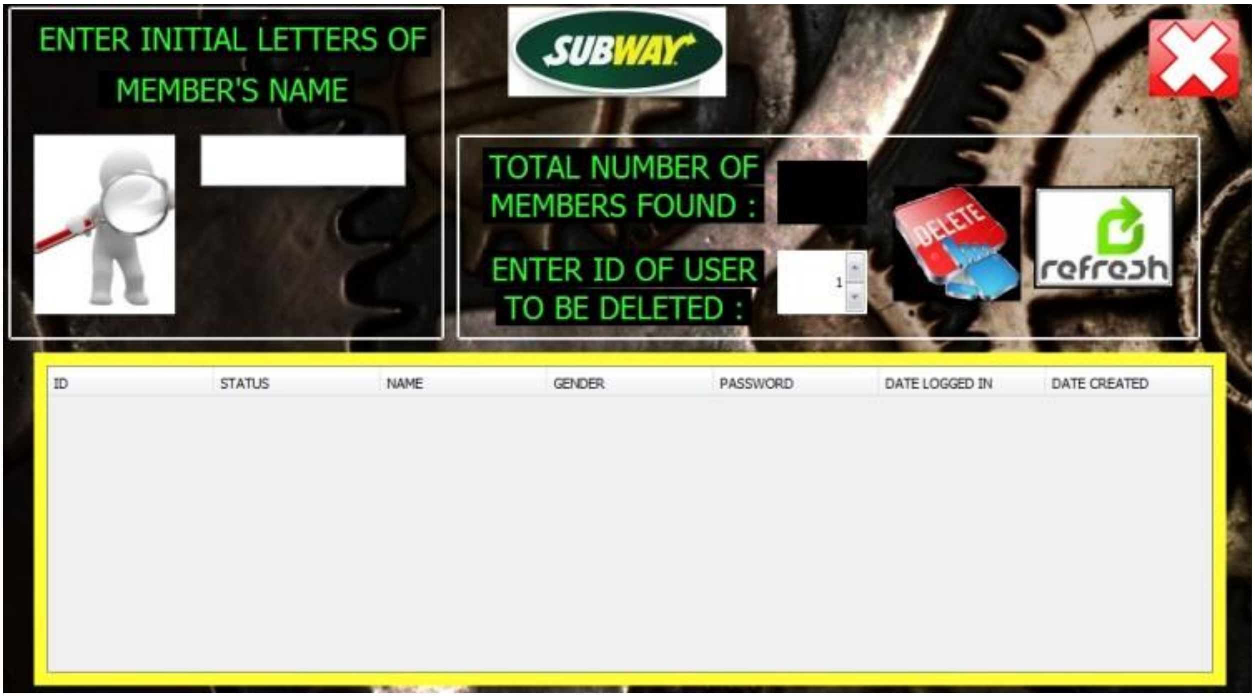
```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{

    new user_dashboard().setVisible(true);

    dispose();

}

private void formWindowGainedFocus(java.awt.event.WindowEvent evt) {

 Calendar c = Calendar.getInstance();
```

```
date.setText("" +c.getTime());

try{

        Class.forName("java.sql.Driver");

Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/
subway","root","123");

Statement stmt=con.createStatement();

Statement stmt1=con.createStatement();


String count_of_records="select count(*) as count from
orders where logged_in='online';";

String orders="select * from orders where logged_in='online';";


ResultSet rs = stmt1.executeQuery(count_of_records);

 while(rs.next()){

    int number = Integer.parseInt(rs.getString("count"));

     if(number==0){

     new user_dashboard().setVisible(true);

   JOptionPane.showMessageDialog(null, "YOU HAVE NOT
PLACED ANY ORDER YET !");

    dispose();

}
```

```
   else{
     rs = stmt.executeQuery(orders);
     int i;
    int sum=0;
 for(i=number;i>=0;i--){
 while(rs.next()){
  String item_ordered = rs.getString("item_ordered");
 double amount=Double.parseDouble(rs.getString("amount"));
double
quantity=Double.parseDouble(rs.getString("quantity"));
  String date_ordered=rs.getString("date_ordered");
      String status=rs.getString("status");
      double total= (amount*quantity);
DefaultTableModel tables = (DefaultTableModel)info.getModel();
Object [] names_found = {item_ordered, amount, quantity,
date_ordered, status,total};
    tables.addRow(names_found);
     sum += total;
}
    total.setText(""+sum+"/-");
    no.setText(""+ info.getRowCount());
   }
  }
 }
}
catch(Exception e){
    JOptionPane.showMessageDialog(null, e.getMessage());
  }
}
```

OWNERS SECTION

Main Frame: Owner User

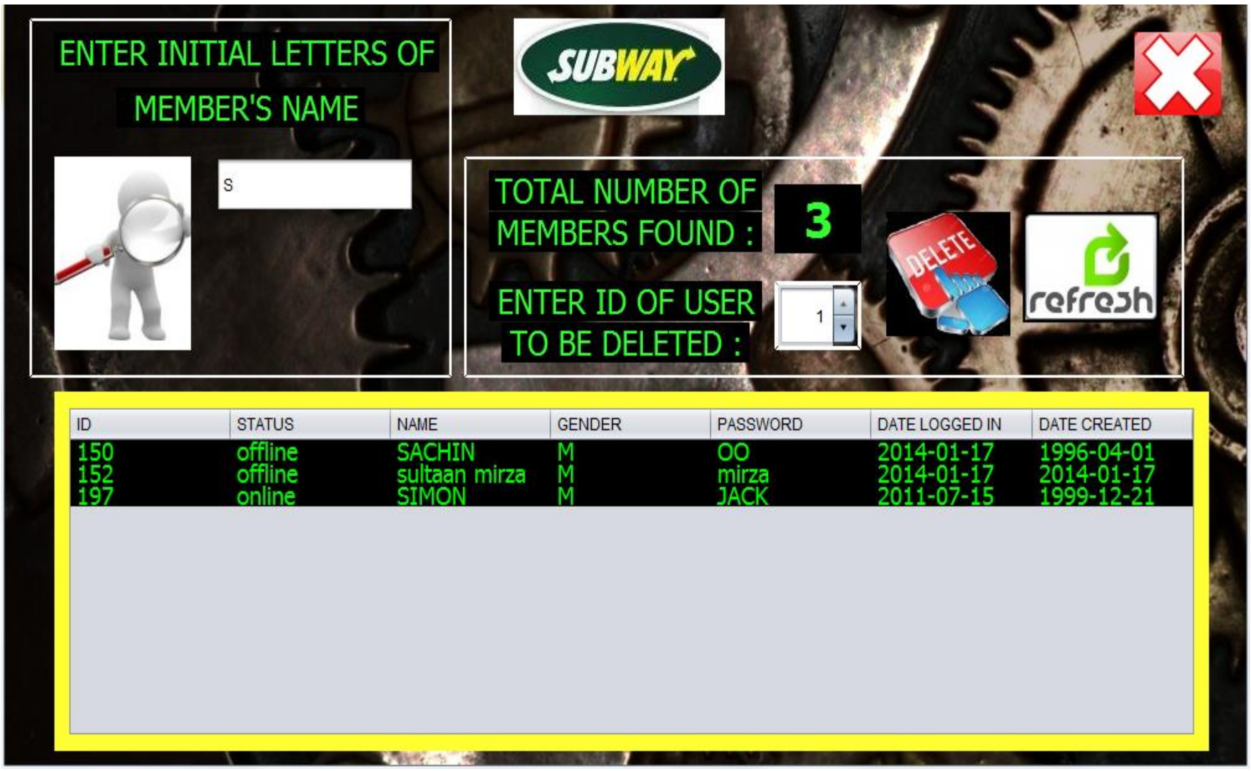

ENTER INITIAL LETTERS OF MEMBER'S NAME
SUBWAY
TOTAL NUMBER OF MEMBERS FOUND :
ENTER ID OF USER TO BE DELETED :
1
DELETE
refresh

ID	STATUS	NAME	GENDER	PASSWORD	DATE LOGGED IN	DATE CREATED

SOURCE CODE: Owner User.java

```
public void search_database(){

String key=(match.getText().toLowerCase());

 try{

     Class.forName("java.sql.Driver");

Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/subway","
root","123");

Statement stmt = con.createStatement();
```

```
String match="select count(*) as total_no_of_members from members where name like '"+key+"%';";

ResultSet rs = stmt.executeQuery(match);

while(rs.next()){

String number1 = rs.getString("total_no_of_members");

int number = Integer.parseInt(number1);

String search = "select * from members where name like '"+key+"%' ;";

Rs = stmt.executeQuery(search);

DefaultTableModel tables =  (DefaultTableModel)tbl.getModel() ;

  int count=tables.getRowCount();

   int m;

  for (m=count; m>0; m--){

      tables.removeRow(0);

 }

    tbl.setModel(tables);

int i;

for(i=number;i>=0;i--){

   while(rs.next()){

    int ID=Integer.parseInt(rs.getString("ID"));

String status=rs.getString("status");

String name=rs.getString("name");

 String gender=rs.getString("gender");

 String password=rs.getString("password");
 String date_last_logged_in=rs.getString("date_last_logged_in");

 String date_created=rs.getString("date_created");

 Object [] names_found = { ID, status, name, gender, password,
date_last_logged_in, date_created };

 DefaultTableModel table =  (DefaultTableModel)tbl.getModel() ;

 table.addRow(names_found);

  }

}

     no.setText(""+ tbl.getRowCount());

  }

}

 catch(Exception e){

  JOptionPane.showMessageDialog(null, e.getMessage());

  }

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{

    new owners_dashboard().setVisible(true);

    dispose();

  }

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{

     int id=(Integer)(sp.getValue());
    if(id==101){

        JOptionPane.showMessageDialog(null, "SORRY!. THIS ACCOUNT
BELONGS TO THE OWNER AND CANNOT  BE DELETED.");

}

 else{

     try {

          Class.forName("java.sql.Driver");

Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/subway","root"
,"123");

String delete_records_from_members="delete from members where ID
="+ id +" ;";

String delete_records_from_contact_info="delete from contact_info
where ID ="+ id +" ;";

String delete records from orders = "delete from orders where ID ="+ id +" ;";


Statement stmt1=con.createStatement();

Statement stmt2=con.createStatement();

Statement stmt3=con.createStatement();


stmt1.executeUpdate(delete_records_from_members);

stmt2.executeUpdate(delete_records_from_contact_info);

stmt3.executeUpdate(delete_records_from_orders);

JOptionPane.showMessageDialog(null, "ACCOUNT DELETED.");

con.close();
}
 catch(Exception e){
          JOptionPane.showMessageDialog(this,e.getMessage());
   }
  }
}
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
 search_database();
}
```



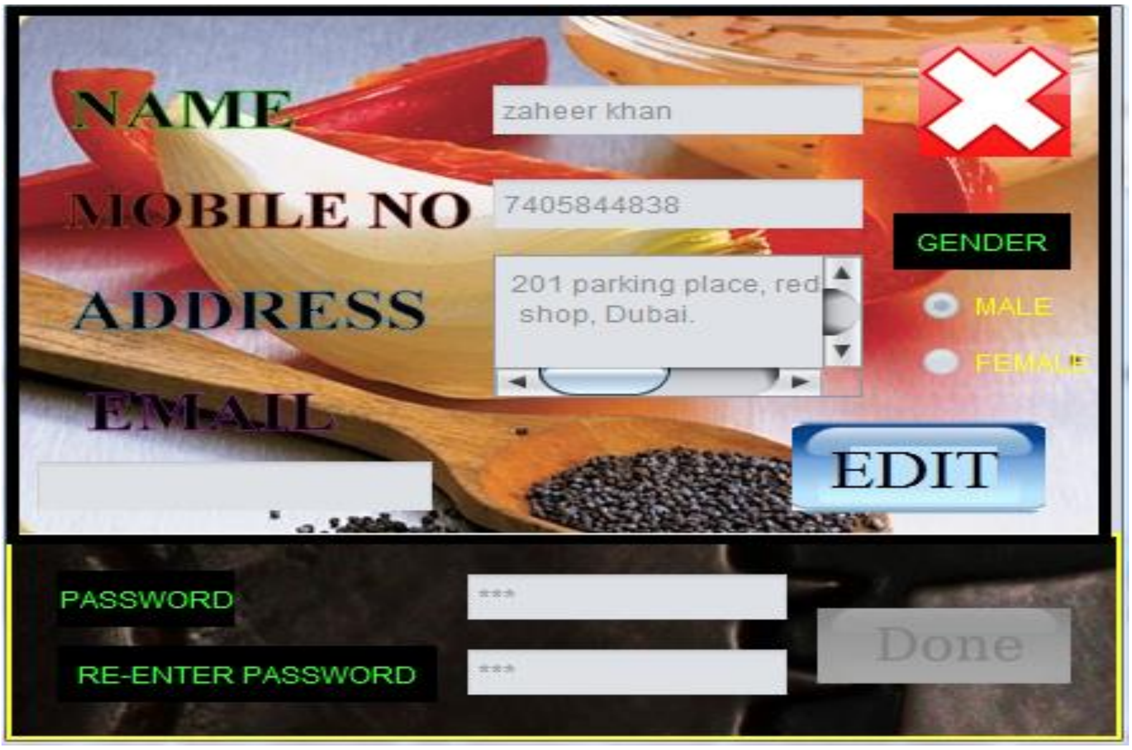
ENTER INITIAL LETTERS OF MEMBER'S NAME S

TOTAL NUMBER OF MEMBERS FOUND : 3

ENTER ID OF USER TO BE DELETED : 1

ID	STATUS	NAME	GENDER	PASSWORD	DATE LOGGED IN	DATE CREATED
150	offline	SACHIN	M	OO	2014-01-17	1996-04-01
152	offline	sultaan mirza	M	mirza	2014-01-17	2014-01-17
197	online	SIMON	M	JACK	2011-07-15	1999-12-21

Main Frame: Owner Profile



NAME zaheer khan

MOBILE NO 7405844838

ADDRESS 201 parking place, red shop, Dubai.

EMAIL

PASSWORD ***

RE-ENTER PASSWORD ***

SOURCE CODE: Owner profile.java

```
private void gActionPerformed(java.awt.event.ActionEvent evt)
{

String name=a.getText().trim().toLowerCase();

String gender="";

 if(e.isSelected())

    gender = "M";

else
```

```
     gender = "F";

String address=c.getText().trim().toLowerCase();

String email_id = d.getText().trim().toLowerCase();

String password= "";

String password1 = new String(h.getPassword());

String password2 = new String(i.getPassword());

       if(password1.equals(password2)){

            password=password1;

     }

      else{

      JOptionPane.showMessageDialog(null, "the password you
entered didn't matched, please try again");

           g.doClick();

    }

       if(name.equals("")||
password.equals("")||email_id.equals("")||address.equals(""))

JOptionPane.showMessageDialog(null, "All fields are mandatory.");

      else{

         try {

            long mobile = Long.parseLong(b.getText());
```

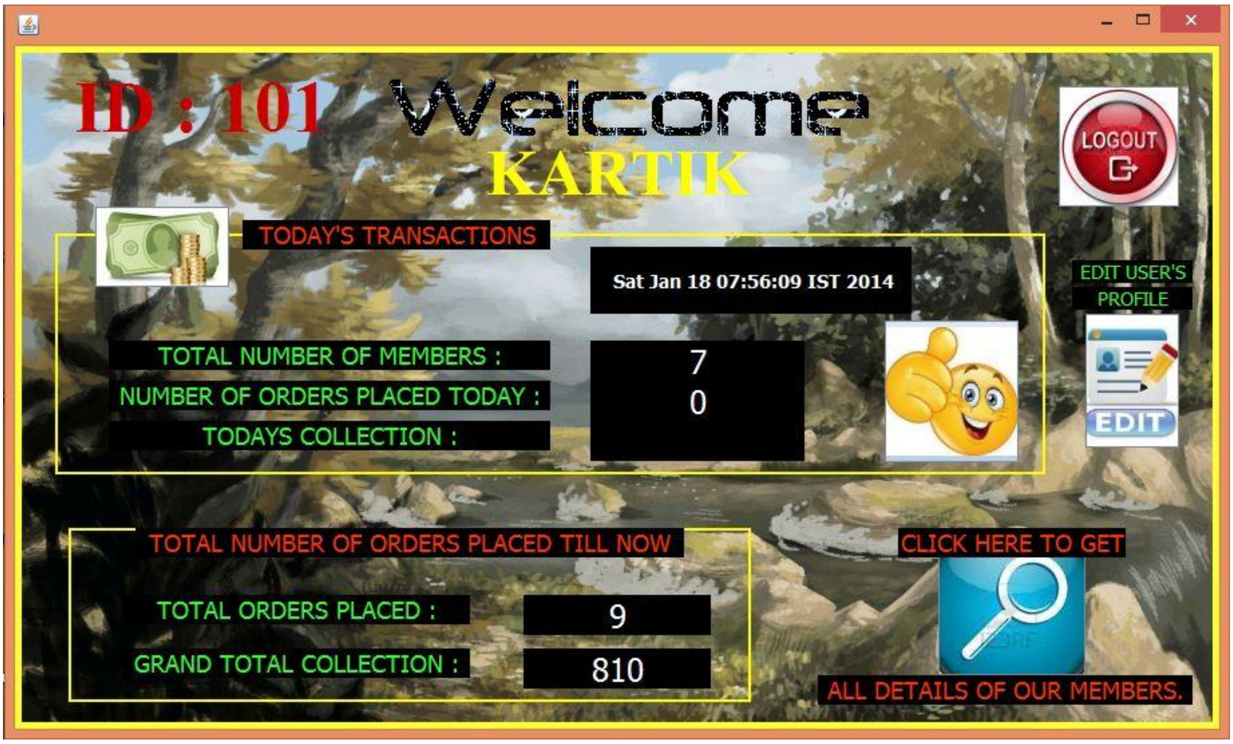
```
          Class.forName("java.sql.Driver");

Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:33
06/subway","root","123");

           Statement stmt=con.createStatement();

           Statement stmt1=con.createStatement();

           Statement stmt2=con.createStatement();

String update_members="update members set name='"+ name +"'
, gender='"+gender+"'  , password='"+password+"'where ID=101 ;";


String update_contact_info= "update contact_info  set name='"+
name +"'  , mobile_no="+mobile+" ,email='"+email_id+"'  ,
address='"+address+"' where ID=101 ;";

String update_orders= "update orders set name='"+ name +"'
where ID=101 ;";

        stmt.executeUpdate(update_members);

        stmt1.executeUpdate(update_contact_info);

        stmt2.executeUpdate(update_orders);

     stmt.close();

     con.close();

JOptionPane.showMessageDialog(null, "changes saved
successfully");
```

```
      new owners_dashboard().setVisible(true);
             dispose();
 }
   catch(Exception e){
JOptionPane.showMessageDialog(this,e.getMessage());
                       dispose();
   }
 }
}
   private void bFocusLost(java.awt.event.FocusEvent evt) {
       String mobile=b.getText().trim();
       if(mobile.equals("")){
  JOptionPane.showMessageDialog(null, "this field is mandatory.");
 }
  else{
   int initial_digit=Integer.parseInt(mobile.substring(0, 1));
      int length=mobile.length();
      if(length==10){
 if(initial_digit==8|| initial_digit==9 || initial_digit==7){
```

```
String duplicate_check="select mobile_no from contact_info
where mobile_no="+mobile+" and status='offline' and ID>101 ;";

try{

          Class.forName("java.sql.Driver");

Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/
subway","root","123");

    Statement stmt=con.createStatement();

    ResultSet rs=stmt.executeQuery(duplicate_check);

   while(rs.next()){

String
mysql_mobno=(rs.getString("mobile_no")).toLowerCase().trim();

   if(mobile.equals(mysql_mobno)){

JOptionPane.showMessageDialog(null, "Sorry! this number
has already been taken by our existing user. please try a
different mobile number.");

                  b.setText("");

       }

    }

catch(Exception e){

     JOptionPane.showMessageDialog(null, e.getMessage());

  }
}
 else{
      JOptionPane.showMessageDialog(null, "This kind of
mobile number does not exist in India. Please enter a VALID
number.");
          b.setText("");
   }
 }
else{
JOptionPane.showMessageDialog(null, "Please enter a 10 digit
number ONLY");
              b.setText("");
     }
  }
}


private void bKeyTyped(java.awt.event.KeyEvent evt) {
       char ch = evt.getKeyChar();
       if( ((ch>='0') && (ch<='9')) ){
          evt.consume();
```

```
        }

    }

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{

        new owners_dashboard().setVisible(true);

        dispose();

    }
```

Main Frame: OWNERS DASHBOARD



Sat Jan 18 07:56:09 IST 2014

TOTAL NUMBER OF MEMBERS : 7

NUMBER OF ORDERS PLACED TODAY : 0

TODAYS COLLECTION :

TOTAL ORDERS PLACED : 9

GRAND TOTAL COLLECTION : 810

SOURCE CODE: Owners Dashboard.java

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt)
{

    new Owners_user__search().setVisible(true);

     dispose();

 }

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt)
{

      new LOGIN_FORM().setVisible(true);
```

```
        dispose();

}

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt)
{

       new edit_owners_profile().setVisible(true);

       dispose();

}

private void formWindowGainedFocus(java.awt.event.WindowEvent evt) {

 Calendar c = Calendar.getInstance()

 datelb.setText(" " +c.getTime());

       try {

              Class.forName("java.sql.Driver");

Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/
subway","root","123");

Statement stmt=con.createStatement();

Statement stmt1=con.createStatement();

Statement stmt2=con.createStatement();


String present_members= "Select count(*) as number from
members;";

String total_transaction="SELECT total_orders , total_money FROM
total_collection ;";
String transaction_done_today="SELECT COUNT(*) as recent_count,
sum(amount) as recent_total  FROM total_collection_perday WHERE
date='"+(c.get(Calendar.YEAR))+"-"+(c.get(Calendar.MONTH)+1)+"-
"+(c.get(Calendar.DATE))+"' ;";

ResultSet rs = stmt.executeQuery(present_members);

while(rs.next()){

    tot_no.setText(rs.getString("number"));

  rs = stmt1.executeQuery(total_transaction);

while(rs.next()){

    tot_orders.setText(rs.getString("total_orders"));

    grand_total.setText(rs.getString("total_money"));

   rs = stmt2.executeQuery(transaction_done_today);

   while(rs.next()){

        no_of_order.setText(rs.getString("recent_count"));

        tot_amt.setText(rs.getString("recent_total"));

    }

 }

}

    catch(Exception e){

  }

}
```

BIBLIOGRAPHY:-

- IP by Sumita Arora
- www.subway.co.in
- www.Wikipedia.com
- www.Google.com