NEW HORIZON
COLLEGE OF ENGINEERING
Autonomous College, Affiliated to VTU | Approved by AICTE New Delhi & UGC
Accredited by NAAC with 'A' Grade & Accredited by NBA

Golden JUBILEE
YEARS OF EXCELLENCE IN EDUCATION

A MINI PROJECT

REPORT

for

Mini Project in JAVA (19CSE48)

VIRTUAL SHOPPING

Submitted by

S YERRINATHA REDDY
1NH19CS233
4-D

In partial fulfillment for the award of
the degree of

Bachelor of Engineering

in

COMPUTER SCIENCE AND ENGINEERING

NEW HORIZON
COLLEGE OF ENGINEERING
Autonomous College, Affiliated to VTU | Approved by AICTE New Delhi & UGC
Accredited by NAAC with 'A' Grade & Accredited by NBA

50 Golden Jubilee
YEARS OF EXCELLENCE IN EDUCATION

Certificate

This is to certify that the mini project work titled

VIRTUAL SHOPPING

Submitted in partial fulfillment of the degree of
Bachelor of Engineering in
Computer Science and Engineering by

S YERRINATHA REDDY
1NH19CS233

DURING
EVEN SEMESTER 2020-2021
for
19CSE48

11/8/21
Signature of Reviewer

Signature of HOD

SEMESTER END EXAMINATION

Name of the Examiner — Signature with date

1. ______________________ ______________________

2. ______________________ ______________________

1NH19CS233 _Yerrinatha Reddy

ORIGINALITY REPORT

13%	%	13%	%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Perry Xiao. "Basic Java Programming", Wiley, 2018 Publication	3%
2	Xin Wu, Yonghui Wang, Lianming Wang. "Synchronization Methods of Multiple High Frame Rate Industrial Cameras Using a General-Purpose Computer", Arabian Journal for Science and Engineering, 2021 Publication	2%
3	Vishal B. Pattanashetty, Nalini C. Iyer, Abhinanadan Dinkar, Supreeta Gudi. "Chapter 50 Self-coordinating Bus Route System to Avoid Bus Bunching", Springer Science and Business Media LLC, 2017 Publication	1%
4	"JDBC Recipes", Springer Science and Business Media LLC, 2005 Publication	1%
5	Amandeep Kaur, Manpreet Kaur. "Analysis of Code Refactoring Impact on Software Quality", MATEC Web of Conferences, 2016 Publication	1%

6	"Beginning Java® Programming", Wiley, 2012 Publication	1%
7	"Chapter 6 Some Useful Classes", Springer Science and Business Media LLC, 2005 Publication	1%
8	Giti Javidi. "chapter 9 Hyperspectral Microwave Atmospheric Sounder (HyMAS) Graphical User Interface Design", IGI Global, 2019 Publication	<1%
9	Ung Mo Kim, Phillip C-Y Sheu. "Database Languages", Wiley, 1999 Publication	<1%
10	Luan P. Lima, Lincoln S. Rocha, Carla I. M. Bezerra, Matheus Paixao. "Assessing exception handling testing practices in open-source libraries", Empirical Software Engineering, 2021 Publication	<1%
11	"Masterkurs Client/Server- Programmierung mit Java", Springer Science and Business Media LLC, 2007 Publication	<1%
12	Laura Caponetti, Giovanna Castellano. "Chapter 5 Java for Image Processing", Springer Science and Business Media LLC, 2017	<1%

	Publication	
13	Rubin H. Landau. "A First Course in Scientific Computing", Walter de Gruyter GmbH, 2005 Publication	<1%
14	"Fundamentals of Computer Organization and Design", Springer Science and Business Media LLC, 2003 Publication	<1%
15	Bernhard Beckert, Steffen Schlager. "Refinement and retrenchment for programming language data types", Formal Aspects of Computing, 2005 Publication	<1%
16	João Fernando Ferreira, João Luís Sobral. "Chapter 21 ParC#: Parallel Computing with C# in .Net", Springer Science and Business Media LLC, 2005 Publication	<1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	Off		

ABSTRACT

Online shopping has thus far tended to be a niche business – highly successful in selling digital products such as shares, software and, increasingly, music and films, it has been lesssuccessful in persuading the purchasers of ‘traditional’ goods such as cars, clothes , toiletries, or household appliances to forsake their physical retailers and move into cyberspace.

In this wide - ranging review paper we investigate the issue of the virtual experience endeavouring to understand what is needed for a successful ‘shopping experience’ online and what the possible obstacles or pitfalls along the way might be. We initially introduce the concepts of virtual presence (the sense of ‘being there’) and virtual reality, discussing the possible roles both can play in providing a solution to the problem of effective online shopping. We then consider the Experience Economy, a concept which encapsulates many of the issues related to the problem of online shopping and which suggests ways in which online retailers can enhance the effectiveness of their sites by means of a virtual ‘experience’.

Having set the scene for online shopping, we discuss eTailing today in terms of direct product experience and the opportunities which cyber-shopping offers to replicate this process. Finally, we identify some of the possibilities and problems of online shopping today , the current status of virtual presence in retailing with two micro-cases of success and failure.

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be impossible without the mention of the people who made it possible, whose constant guidance and encouragement crowned our efforts with success..

I have great pleasure in expressing gratitude to **Dr. Mohan Manghnani**, Chairman of New Horizon Educational Institutions for providing necessary infrastructure and creating good environment.

I take this opportunity to express my profound gratitude to **Dr. Manjunatha ,** Principal, New Horizon College of Engineering, for his constant support and encouragement.

I am grateful to **Dr. Amarjeet Singh,** Dean - Academics, for his unfailing encouragement and suggestions, given to me in the course of my project work.

I would also like to thank **Dr. B. Rajalakshmi**, Professor and Head, Department of Computer Science and Engineering, for her constant support.

I also express my gratitude to **Ms. Pramilarani K**, Senior Assistant Professor, Department of Computer Science and Engineering, my project guide, for constantly monitoring the development of the project and setting up precise deadlines. Her valuable suggestions were the motivating factors in completing the work.

S YERRINATHA REDDY

1NH19CS233

CONTENTS

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Online shopping system is a process in which people (specific customers) are being provided with the option of purchasing goods and services directly from the seller, all in real time environment. Online shopping is an application of internet as electronic commerce.

This project is a web based shopping system for an existing shop. The project objective is to deliver the online shopping application into android platform. Online shopping is the process whereby consumers directly buy goods or services from a seller in real-time, without an intermediary service, over the Internet. It is a form of electronic commerce. This project is an attempt to provide the advantages of online shopping to customers of a real shop. It helps buying the products in the shop anywhere through internet by using an android device. Thus the customer will get the service of online shopping and home delivery from his favorite shop.

After all this process the payment method will be appeared the payment should also be done in online for example the payment can done through Paytm , Google pay , PhonePe ...etc. After the payment the products will be booked with the given details.

After all this process the payment method will be appeared the payment should also be done in online for example the payment can done through net banking, cards etc. After the payment the products will be booked with the given details.

The main scope of this project is to transver a lot of areas ranging from business concepts and required to perform various options to the customer it is helpful to the Customers tasks whenever customer to buy a product this project is helpful for both customer and seller.

1.1 PROBLEM DEFINITION:

Online shopping is a process that has been well known only in the last few years to be precise. However, it focuses on consumers choosing their products and buying them online.

The entire process is virtually done and there is no need for the person to be physically present at the place at the moment. It is an extensively fuss - free process which has saved a lot of precious time of people who are busy with their daily life.

Online shopping project deals with all these online shopping sites and their proper functioning such that people can access it easily . However online shopping poses different problems.

The quality of the product which we see online and the product we receive is often different. This creates a bad impression for the consumer. Also, delivery time may vary in the case of online shopping

ADVANTAGES OF VIRTUAL SHOPPING:

1. Quick and easy access to customer for buying products.
2. Money can be paid online.
3. Will be helpful in emergency cases.
4. Cancellation is easy to customer.
5. Secure.
6. setup an price from anywhere
7. Discount rates.
8. Easy usage to the customer.
9. Customer can save their time.

1.2. OBJECTIVES:

The main objective of the Online Shopping System project is to help the ongoing user help to attain an easy way to navigate the customer details and solve the offline store problems. It is basically a very instant processing System by which customers can get the product in the right time.

Company is able to improve their customer satisfaction level according to their number of bookings , increasing efficiency by provides better services to their customer. The proposed web-based system has the following features that will be included in the Online Shopping System.

1.3 METHODOLOGY TO FOLLOWED :

In this project we use OOP's with java concepts as a main topic to determine waiting time of customer in a queue using token number.

In this project first we need to create a class or oops concepts for online shopping application. First, we need to enter our details and we should select type of payment in which customer it's a customer choice.

Using oops with java concept in this project we can able to get the required Online Shopping.

1.4. EXPECTED OUTCOMES:

We run the project as java application.

1. Personal Details of the customer.
2. filling the address .
3. Selecting the product.
4. Specifying the number of days .
5. Calculating the amount.
6. Paying the amount.

You have successfully booked the product and will be delivered.

1.5. HARDWARE AND SOFTWARE REQUIREMENTS

1.5.1 HARDWARE REQUIREMENTS:

Processor:	Inte1 core i3/i4/i5/i6/i7
Speed:	3 GHz to 4 GHz
RAM:	512 MB RAM or more.
ROM:	2 GB DDR3

1.5.2 SOFTWARE REQUIREMENTS:

- Jdk installer by Oracle.
- Eclipse IDE for Enterprise java Developers.
- Windows XP/Windows 7/Windows 8/Windows 10.

1.5.3 Eclipse IDE.

In the context of computing, Eclipse is an integrated development environment (IDE) for developing applications using the Java programming language and other programming languages such as C/C++, Python, PERL, Ruby etc. The Eclipse platform which provides the foundation for the Eclipse IDE is compose of plug-ins and is designed to be extensible using additional plug-ins. Developed using Java, the Eclipse platform can be used to develop rich client applications, integrated development Environments and other tools. Eclipse can be used as an IDE for any programming language for which a plug-in is available. The Java Development Tools (JDT) project provides a plug-in that allows Eclipse to be used as Java IDE, is a plug in that allows Eclipse to be used as a Python IDE, C/C++ Development Tools (CDT) is a plug- in that allows Eclipse to be used for developing application using JAVA, The Eclipse Scale plug-in allows Eclipse to be used an IDE to develop Scale applications and is a plug-in to eclipse that provides complete development tool for PHP.

Eclipse uses plug-ins to provide all the functionality within and on top of the run-time system. Its run-time system is based on implementation of the core framework specification. In the addition to allowing the Eclipse Platform to be extended using other programming languages, such as C and Python, the plug-in framework allows the Eclipse Platform to work with the typesetting languages like and networking applications such as and database management systems. The plug-in architecture supports writing any desired extension to the environment, such as for configuration management. Java and CVS support is provided in the Eclipse SDK, with support for other version control systems provided by third-party plug-ins.

With the exception of a small run-time kernel, everything in Eclipse is a plug-in. Thus, every plug-in developed integrates with Eclipse in the same way as other plug-ins; in this respect, all features are created equal Eclipse provides plug-ins for a wide variety of features, some of which are from third parties using both free and commercial models. Examples of plug-ins include for Unified Modeling Language (UML), for Sequence and other UML diagrams, a plug-in for DB Explorer, and many more.

CHAPTER 2

OBJECT ORIENTED CONCEPTS

2.1 CLASS

A class is a user defined blueprint or prototype from which objects are created. It represents the set of properties or methods that are common to all objects of one type. In general, class declarations can include these components, in order:

Modifiers : A class can be public or has default access.

Class name: The name should begin with an initial letter (capitalized by convention).

Super class: The name of the class's parent (super class), if any, preceded by the keyword extends. A class can only extend (subclass) one parent.

Body: The class body surrounded by braces, { }.

Example:

```
class AM {
        System.out.print.ln("hello world");
}
```

2.2 CONSTRUCTORS:

Constructors are used for initializing new objects. Fields are variables that provides the state of the class and its objects, and methods are used to implement the behavior of the class and its objects. There are various types of classes that are used in real time applications such as nested classes, lambda expressions.

2.3 OBJECTS:

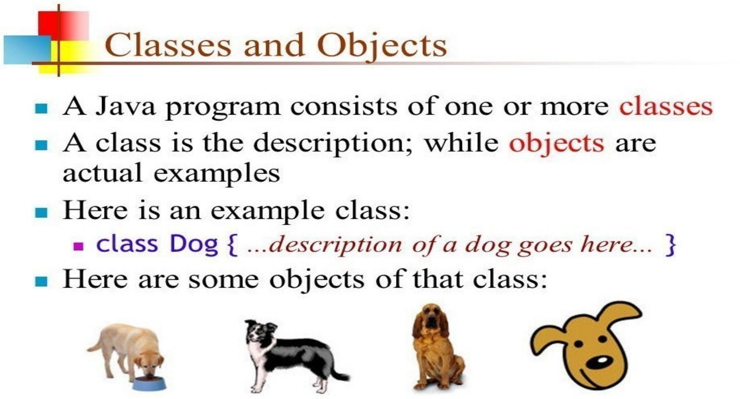
An entity that has state and behavior is known as an object e.g., chair, bike, marker, pen, table, car, etc. It can be physical or logical (tangible and intangible). The example of an intangible object is the banking system.

An object has three characteristics:

- State: represents the data (value) of an object.

- Behavior: represents the behavior (functionality) of an object such as deposit, withdraw, etc.

- Identity: An object identity is typically implemented via a unique ID. The value of the ID is not visible to the external user. However, it is used internally by the JVM to identify each object uniquely.



Classes and Objects

- A Java program consists of one or more classes
- A class is the description; while objects are actual examples
- Here is an example class:
  - class Dog { *...description of a dog goes here...* }
- Here are some objects of that class:

Example:

```
public class P

{

public static void main(String [] arg)

{

h ob=new h();

ob.m();

obj.random();      (This how we create an object)
```

```
        }}
```

2.4 INHERITANCE:

Inheritance in Java is a mechanism in which one object acquires all the properties and behaviors of a parent object. ... The idea behind inheritance in Java is that you can create new classes that are built upon existing classes. When you inherit from an existing class, you can reuse methods and fields of the parent class.

- Single Inheritance.
- Multiple Inheritance (Through Interface)
- Multilevel Inheritance.
- Hierarchical Inheritance.
- Hybrid Inheritance (Through Interface)

Example:

```
class h extends q

{

         void m()

        {

        int f,x,r;

        //System.out.println("ENTER CATEGORY\n");

        //Scanner ch=new Scanner(System.in);

        //System.out.println("SELECT THE PRODUCT YOU LIKE TO BUY\n");

        // r=ch.nextInt();
```

In the above we used extends because to enable the inheritance we need to extends keyword

2.5 JAVA BUZZWORDS

- **Simple**

Java is anything but difficult to learn and its language structure is very straightforward, perfect and straightforward. The confounding and uncertain ideas of C++ are either forgotten in Java or they have been re-executed in a cleaner way. Using equal tool, we can evaluate any type of SQL query and display the result on console Scoop equal tool can be applicable for both modeling and defining the SQL statements. That means, we can use equal for insert statements too. If the command executes successfully.

- **Object Arranged**

In java everything is Item which has a few information and conduct. Java can be effectively stretched out as it depends on Article Model. Object-arranged programming (Uh oh) is an

approach that disentangles programming improvement and support by giving a few tenets.

- **Robust**

Java attempts to wipe out mistake inclined codes by underscoring principally on aggregate time blunder checking and runtime checking. However, the primary territories which Java improved were Memory The executives and misused Special cases by presenting programmed City worker and Exemption Dealing with. Approach that disentangles programming improvement and support by giving a few tenets.

- **Platform Free**

On assemblage Java program is accumulated into byte code. This byte code is stage autonomous and can be kept running on

any machine, in addition to this byte code group likewise give security. Any machine with Java Runtime Condition can run Java Projects.

- **Secure**

Java secure highlights it empower us to create infection free, temper free framework. Java program dependably keeps running in Java runtime condition with practically invalid cooperation with framework OS, thus it is progressively secure.

- **Architectural Impartial**

Compiler produces byte codes, which have nothing to do with specific PC design; henceforth a Java program is anything but difficult to decipher on any machine.

•Linked-Rundown

A connected rundown is a straight information structure where every component is a different item. Connected rundown components are not put away at adjacent area; the components are connected utilizing pointers. Every hub of a rundown is comprised of two things - the information and a reference to the following hub. The last hub has a reference to invalid.

Java language and Java Virtual Machine helped in achieving the goal of “write once; run anywhere, any time, forever.”

Changes and upgrades in operating systems, processors and system resources will not force any changes in Java Programs.

Java Provides a way to download programs dynamically to all the various types of platforms connected to the Internet.

It helps in generating Portable executable code.

High Performance

Java performance is high because of the use of byte code.

The byte code was used, so that it was easily translated into native machine code.

2.6. POLYMORPHISM

Polymorphism refers to the ability of OOPs programming languages to differentiate between entities with the same name efficiently. This is done by Java with the help of the signature and declaration of these entities.

For example!

```
 // Java program to demonstrate Polymorphism

// This class will contain

//3 methodsvGthsamename,

// yet the program will

//compile& run successfully public class
Sum (

 // Overloaded sum().

 //This sum takes two int parameters public int
 sum (int x, int y)

  return (x + y);

 // Overloaded sum().

 //This sum takes three int parameters public int
 sum(Int x, Int y, Int z)

  return (x + y + z);

 // Overloaded sum().

 // This sum takes two double parameters
```

```
public double sum( doublex , doubley )

  return (x + y);


publicstaticvoidmain(Stringargs[])        Sum=new
  Sum();

            System.out.println(s.sum(12, 32));

            System.out.println(s.sum(20, 40, 30));

            System.out.println(s.sum(10, 20.5));
```

Ouput:

```
44

90

30.5
```

2.7. EXCEPTION HANDLING:

An exception is an unwanted unexpected event, which occurs during the execution of a program. Eat run time , that disrupts the normal flow of the program's instructions .Exception Hierarchy! All exception and errors types are subclasses of class Throw able, which is case class of hierarchy. One branch is headed by Exception. This class is used for exceptional conditions that user programs should catch. Null Pointer Exception is an example of such an exception. Another branch, Error are used by the Java run-time system (JVM) to indicate errors having to do with the run-time environment itself (JRE).Stack Overflow Error is an example of such an error.

2.8 MULTITHREADING:

computer architecture, multithreading is the ability of a central processing unit (CPU) (or a single core in a multi-core processor to provide multiple threads of execution concurrently, supported by the operating system. This approach differs from multiprocessing. In a multithreaded application, the threads share the resources of a single or multiple cores, which include the computing units, the CPU caches, and the translation look aside buffer.

Where multiprocessing systems include multiple complete processing units in one or more cores, multithreading aims to increase utilization of a single core by using thread-leve parallelism, as well as instruction-level parallelism. As the two techniques are complementary, they are sometimes combined in systems with multiple multithreading CPUs and with CPUs with multiple multithreading cores.

A major area of research is the thread scheduler that must quickly choose from among the list of ready-to-run threads to execute next, as well as maintain the ready-to-run and stalled thread lists. An important subtopic is the different thread priority schemes that can be used by the scheduler. The thread scheduler might be implemented totally in software, totally in hardware, or as a hardware/software combination.

Another area of research is what type of events should cause a thread switch: cache misses, inter-thread communication, DMA completion, etc.

If the multithreading scheme replicates all of the software-visible state, including privileged control registers and TLBs, then it enables virtual machines to be created for each thread. This allows each thread to run its own operating system on the same processor. On the other hand, if only user-mode state is saved, then less hardware is required, which would allow more threads to be active at one time for the same die area or cost.

2.9 JAVA PACKAGE:

Advantage of Java Package:

1) Java package is used to categorize the classes and interfaces so that they can be easily maintained.

2) Java package provides access protection.

3) Java package removes naming collision.

packages are usually defined using a hierarchical naming pattern, with some levels in the hierarchy separated by periods (., pronounced "dot"). Although packages lower in the naming hierarchy are often referred to as "sub packages" of the corresponding packages higher in the hierarchy, there is almost no semantic relationship between packages. The Java Language Specification establishes package naming conventions to avoid the possibility of two published packages having the same name. The naming conventions describe how to create unique package names, so that packages that are widely distributed will have unique namespaces. This allows packages to be separately, easily and automatically installed and catalogued.

In general, a package name begins with the top level domain name of the organization and then the organization's domain and then any sub domains, listed in reverse order. The organization can then choose a specific name for its package. Subsequent components of the package name vary according to an organization's own internal naming conventions.

For example, if an organization in Canada called My Soft creates a package to deal with fractions, naming the package distinguishes the fractions package from another similar package created by another company. If a German company named My Soft also creates a fractions package, but names it fractions, then the classes in these two packages are defined in a unique and separate namespace.

Public members and classes are visible everywhere and private members are visible only in the same class. Classes within a package can access classes and members declared with access.

CHAPTER 3

DESIGN

3.1. DESIGN GOALS:

- Creating a class named as roy. And asking customer their enter their name

```
public class  Roy {
void out()
{
System.out.println("Enter your name");
```

- In this module we create object to access the class. In this we use public static void because it is main class

```
Public    class P{
public static void main(String [] arg)       {
h ob=new h();
ob.m();
//       obj.random();
}
```

- In this module we use java in built functions such as java util scanner. And scanner function is used for user input. And inhertance topic is also used here.

```
import java.util.Scanner;
//import java.math.*;
Class q  {

int m,n;
int r;
//       int m,n=54321;
Object math;
void push()
{
```

```
int a,y,c=0,l=0;
class h extends q   {
void m()
{
int f,x,r;
//System.out.println("ENTER CATEGORY\n");
//Scanner ch=new Scanner(System.in);
//System.out.println("SELECT THE PRODUCT YOU LIKE TO BUY\n");
// r=ch.nextInt();
```

- In this module we uses if else statement for payemts. And we use system.out.println command to print the statement.

```
if(top>r){
System.out.println("MAX NO OF PRODUCTS REACHED\n");
}
else {
Scanner o=new Scanner (System. in);
System. Out . print ln ("PAYMENT OPTION\n");
System .out. print ln ("1.GOOGLE PAY\n2.PHONE PAY\n3.paytm\n");
```

- In this case we use while loop to generating OTP. And this case we use random variable to generate different OTP. And we use if else statement to verify OTP if entered is right or wrong if it is right then booking will be completed

```
while(ii==1){
n=12456;m=(int)(n*Math.random()+12*Math.random());
y=m;n+=123;
System.out.println("YOUR OTP IS :"+y);
top++;
Scanner ch=new Scanner(System.in);
System.out.println("\n enter ur OTP\n");
```

```
int d=ch.nextInt();
if(y==d){
System.out.println("\t\t****************************\n");
System.out.println("\t\tYOUR PRODUCT IS BOOKED \n");
System.out.println("\t\t****************************\n");
ii=0;
break;
continue;
```

3.2. ALOGARITHM:

1. First user need to create an class by using objected oriented concepts.
2. Classes can consist any topics which were related to object oriented concepts

   And object should be created for calling the class.

   3. First customer should enter his username and password customer needed. user is using java in built functions like java.util .

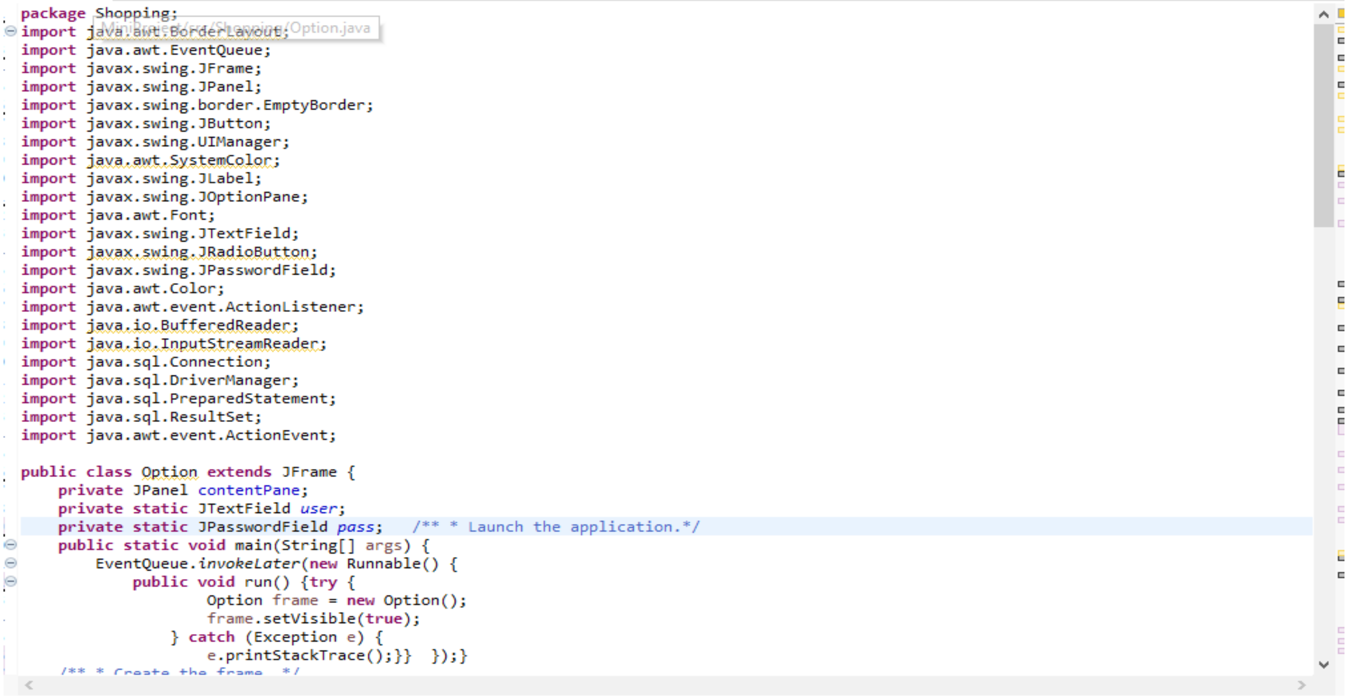
   4. After that customer has to select the product according to their price for the user uses the do while loop for selections

- And switch case is also used for the product selection process
- After selecting the products payment options will displayed by using print statemens and if else statements are for payments

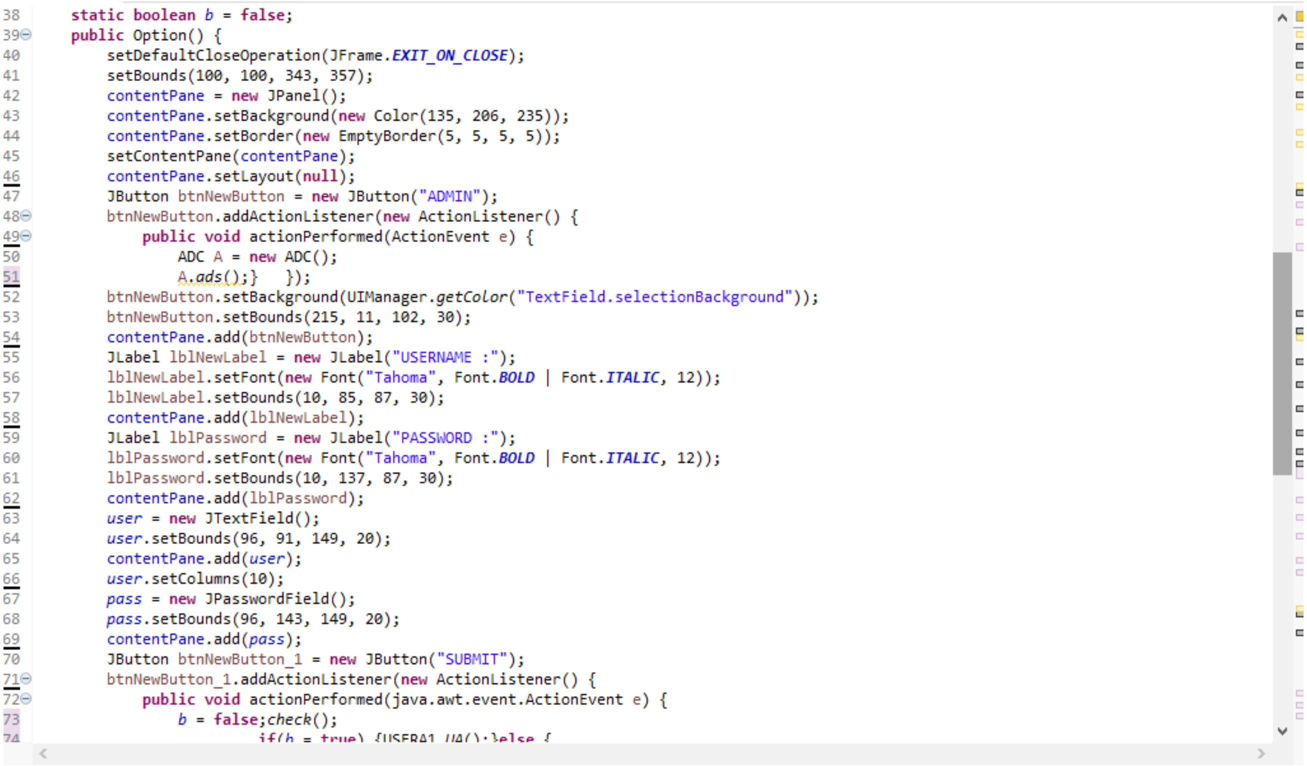
3. There will be five payments options will be displayed on the screen we select and customer should complete their payment
4. After completion of payment OTP will generated customer need to enter the OTP if OTP is verified then booking will be completed

CHAPTER 4

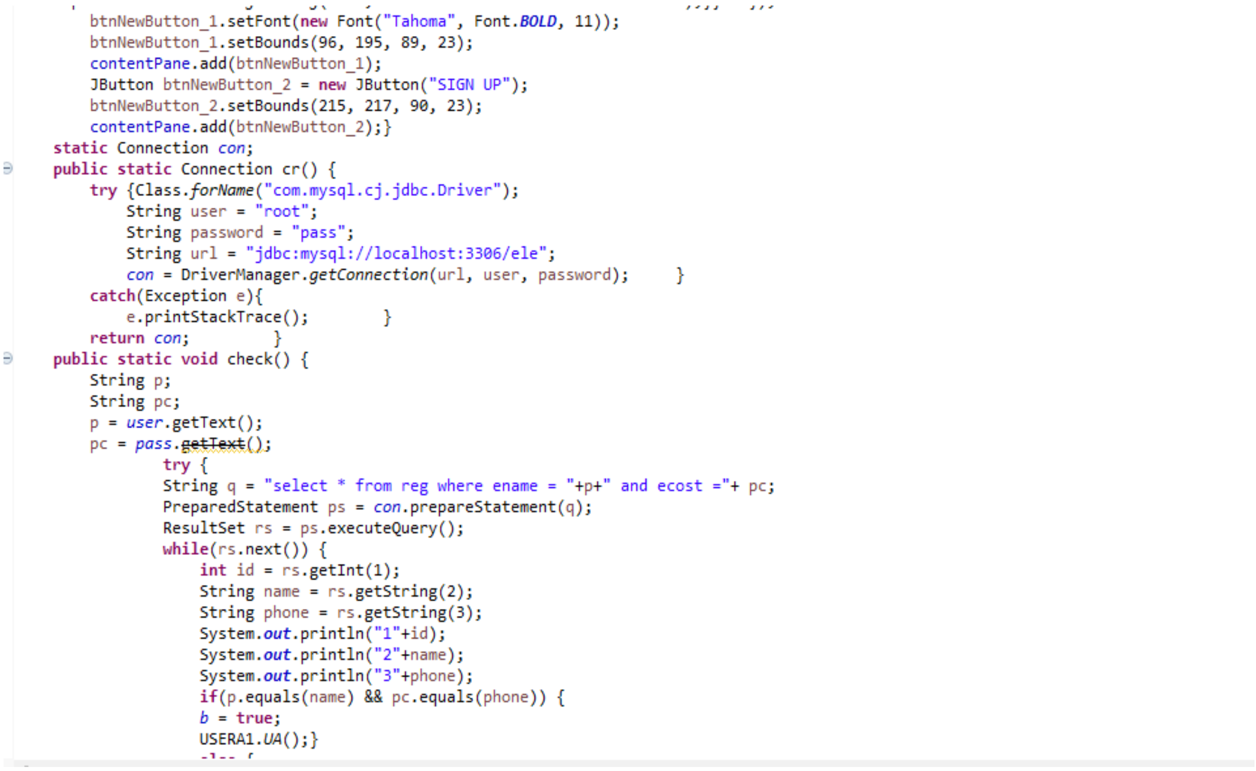
IMPLEMENTATION:



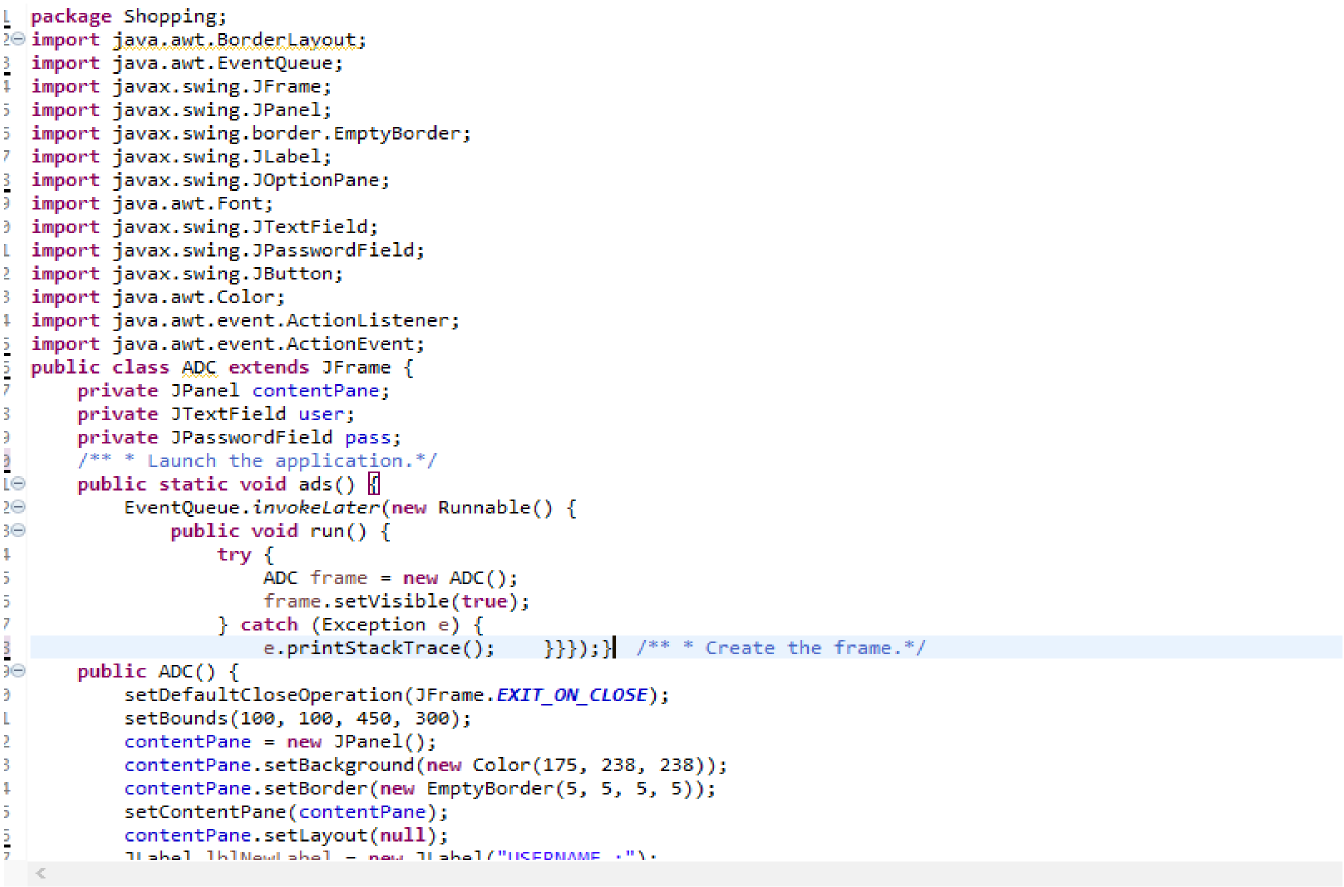
```
package Shopping;
import java.awt.BorderLayout;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.JButton;
import javax.swing.UIManager;
import java.awt.SystemColor;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import java.awt.Font;
import javax.swing.JTextField;
import javax.swing.JRadioButton;
import javax.swing.JPasswordField;
import java.awt.Color;
import java.awt.event.ActionListener;
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.awt.event.ActionEvent;

public class Option extends JFrame {
    private JPanel contentPane;
    private static JTextField user;
    private static JPasswordField pass;   /** * Launch the application.*/
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {try {
                    Option frame = new Option();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();}}  });}
    /** * Create the frame. */
```



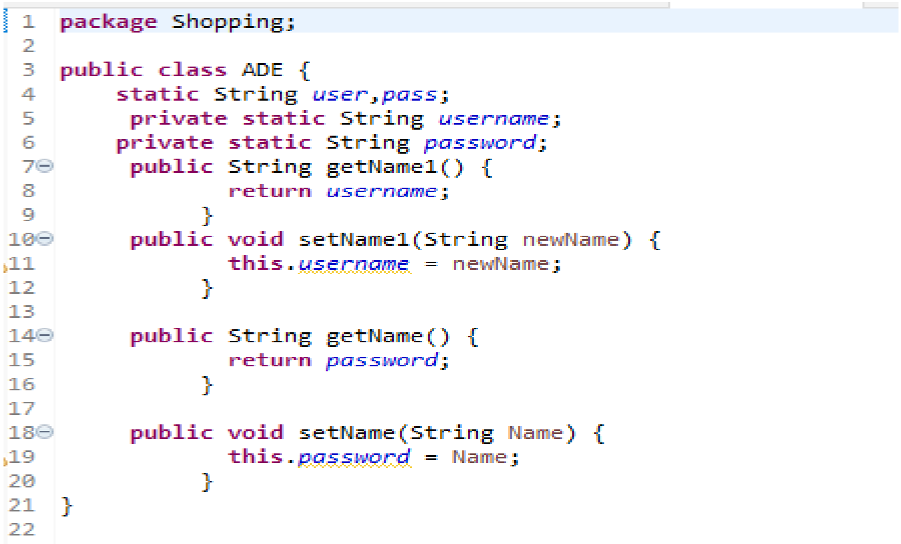
```
    static boolean b = false;
    public Option() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 343, 357);
        contentPane = new JPanel();
        contentPane.setBackground(new Color(135, 206, 235));
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);
        JButton btnNewButton = new JButton("ADMIN");
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                ADC A = new ADC();
                A.ads();}  });
        btnNewButton.setBackground(UIManager.getColor("TextField.selectionBackground"));
        btnNewButton.setBounds(215, 11, 102, 30);
        contentPane.add(btnNewButton);
        JLabel lblNewLabel = new JLabel("USERNAME :");
        lblNewLabel.setFont(new Font("Tahoma", Font.BOLD | Font.ITALIC, 12));
        lblNewLabel.setBounds(10, 85, 87, 30);
        contentPane.add(lblNewLabel);
        JLabel lblPassword = new JLabel("PASSWORD :");
        lblPassword.setFont(new Font("Tahoma", Font.BOLD | Font.ITALIC, 12));
        lblPassword.setBounds(10, 137, 87, 30);
        contentPane.add(lblPassword);
        user = new JTextField();
        user.setBounds(96, 91, 149, 20);
        contentPane.add(user);
        user.setColumns(10);
        pass = new JPasswordField();
        pass.setBounds(96, 143, 149, 20);
        contentPane.add(pass);
        JButton btnNewButton_1 = new JButton("SUBMIT");
        btnNewButton_1.addActionListener(new ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent e) {
                b = false;check();
                    if(b = true) {USERA1_UA();}else {
```



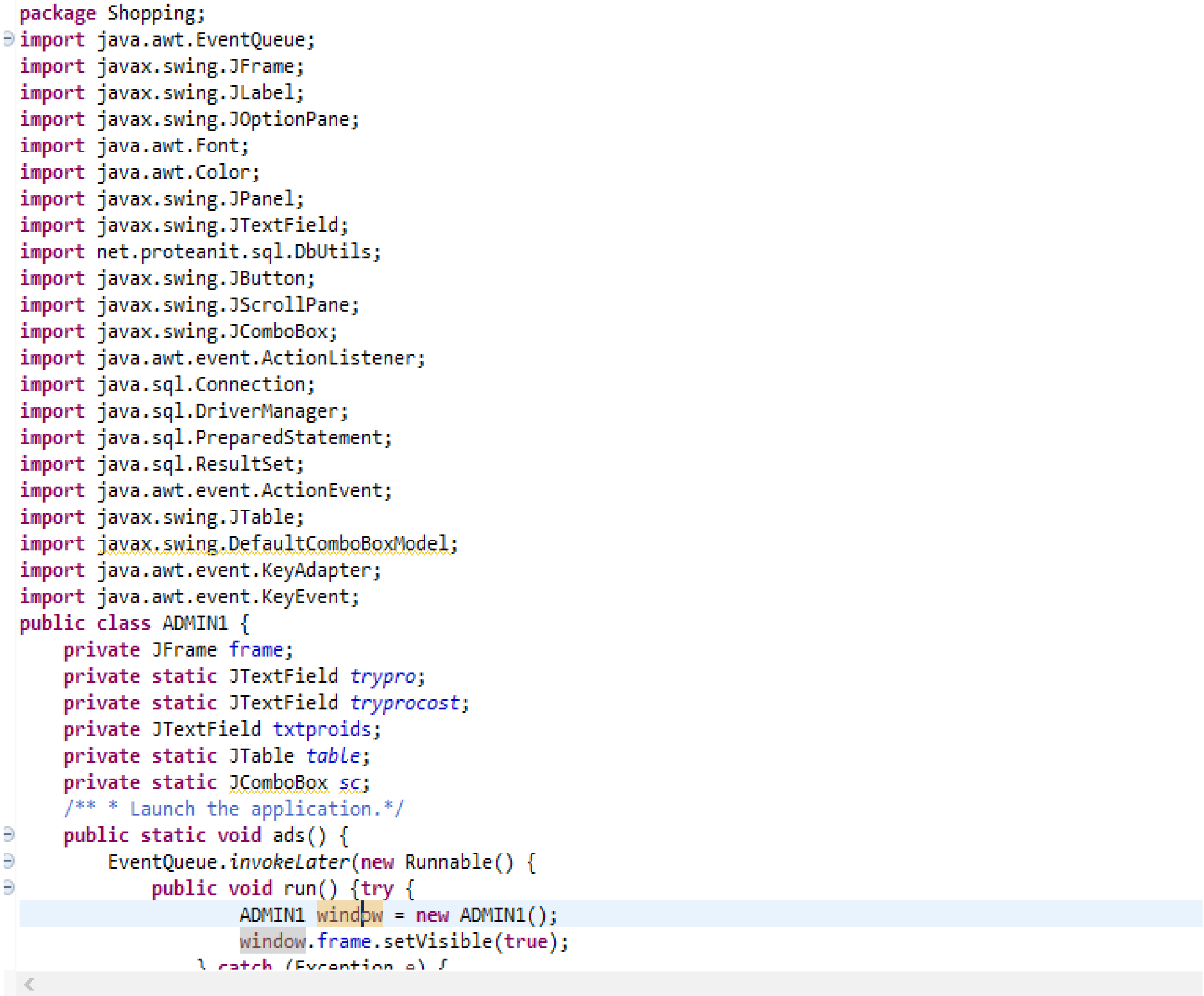
```
        btnNewButton_1.setFont(new Font("Tahoma", Font.BOLD, 11));
        btnNewButton_1.setBounds(96, 195, 89, 23);
        contentPane.add(btnNewButton_1);
        JButton btnNewButton_2 = new JButton("SIGN UP");
        btnNewButton_2.setBounds(215, 217, 90, 23);
        contentPane.add(btnNewButton_2);}
    static Connection con;
    public static Connection cr() {
        try {Class.forName("com.mysql.cj.jdbc.Driver");
            String user = "root";
            String password = "pass";
            String url = "jdbc:mysql://localhost:3306/ele";
            con = DriverManager.getConnection(url, user, password);     }
        catch(Exception e){
            e.printStackTrace();        }
        return con;         }
    public static void check() {
        String p;
        String pc;
        p = user.getText();
        pc = pass.getText();
                try {
                String q = "select * from reg where ename = "+p+" and ecost ="+ pc;
                PreparedStatement ps = con.prepareStatement(q);
                ResultSet rs = ps.executeQuery();
                while(rs.next()) {
                    int id = rs.getInt(1);
                    String name = rs.getString(2);
                    String phone = rs.getString(3);
                    System.out.println("1"+id);
                    System.out.println("2"+name);
                    System.out.println("3"+phone);
                    if(p.equals(name) && pc.equals(phone)) {
                    b = true;
                    USERA1.UA();}
                    else {
```



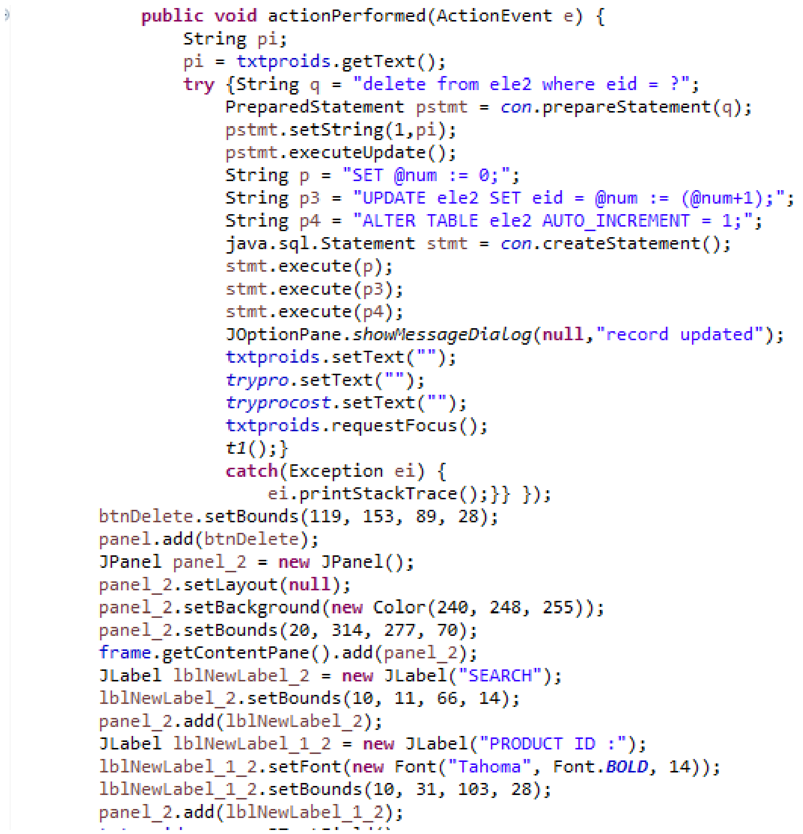
```
package Shopping;
import java.awt.BorderLayout;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import java.awt.Font;
import javax.swing.JTextField;
import javax.swing.JPasswordField;
import javax.swing.JButton;
import java.awt.Color;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
public class ADC extends JFrame {
    private JPanel contentPane;
    private JTextField user;
    private JPasswordField pass;
    /** * Launch the application.*/
    public static void ads() {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    ADC frame = new ADC();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();    }}});}  /** * Create the frame.*/
    public ADC() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 450, 300);
        contentPane = new JPanel();
        contentPane.setBackground(new Color(175, 238, 238));
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);
        JLabel lblNewLabel = new JLabel("USERNAME :");
```



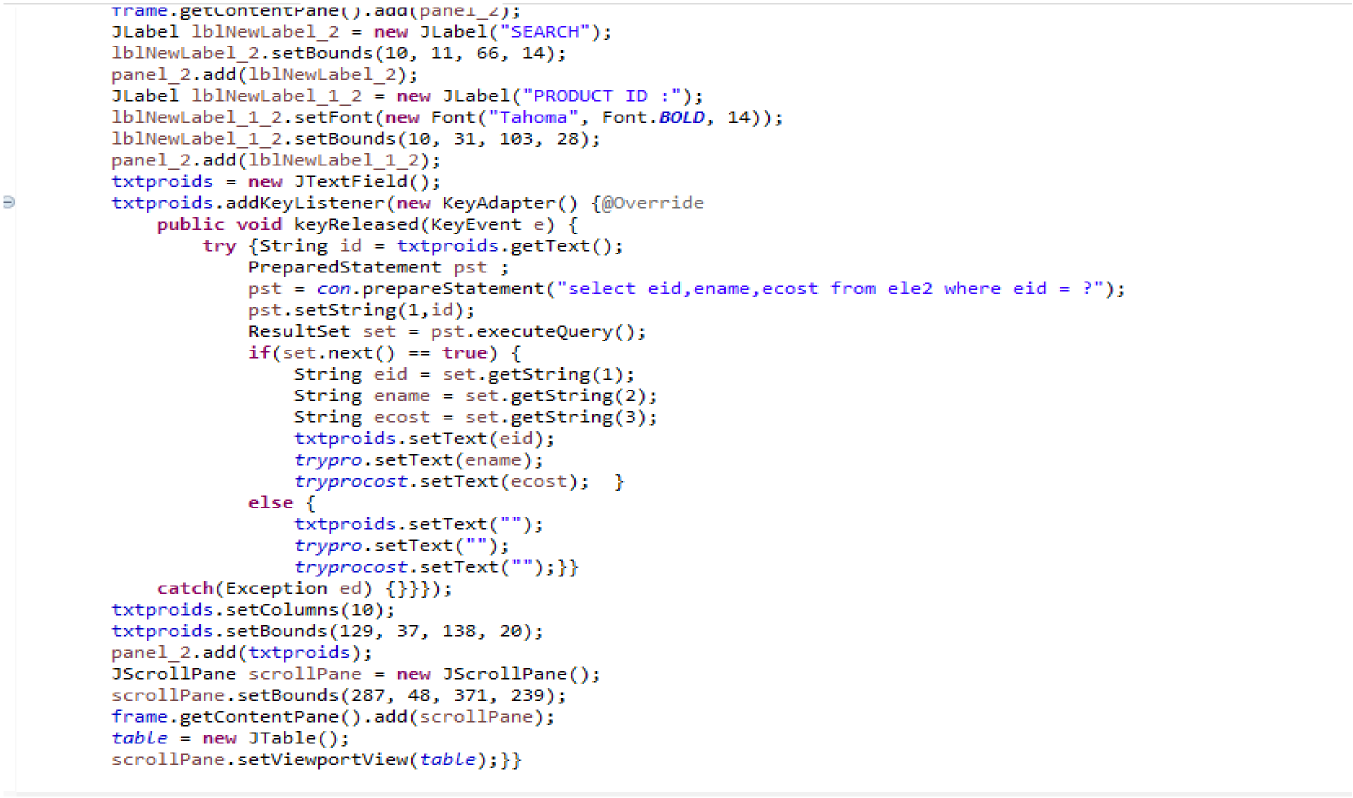
```
package Shopping;

public class ADE {
    static String user,pass;
     private static String username;
    private static String password;
     public String getName1() {
            return username;
          }
     public void setName1(String newName) {
            this.username = newName;
          }

     public String getName() {
            return password;
          }

     public void setName(String Name) {
            this.password = Name;
          }
}

```



```
package Shopping;
import java.awt.EventQueue;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import java.awt.Font;
import java.awt.Color;
import javax.swing.JPanel;
import javax.swing.JTextField;
import net.proteanit.sql.DbUtils;
import javax.swing.JButton;
import javax.swing.JScrollPane;
import javax.swing.JComboBox;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.awt.event.ActionEvent;
import javax.swing.JTable;
import javax.swing.DefaultComboBoxModel;
import java.awt.event.KeyAdapter;
import java.awt.event.KeyEvent;
public class ADMIN1 {
    private JFrame frame;
    private static JTextField trypro;
    private static JTextField tryprocost;
    private JTextField txtproids;
    private static JTable table;
    private static JComboBox sc;
    /** * Launch the application.*/
    public static void ads() {
        EventQueue.invokeLater(new Runnable() {
            public void run() {try {
                    ADMIN1 window = new ADMIN1();
                    window.frame.setVisible(true);
                } catch (Exception e) {
```

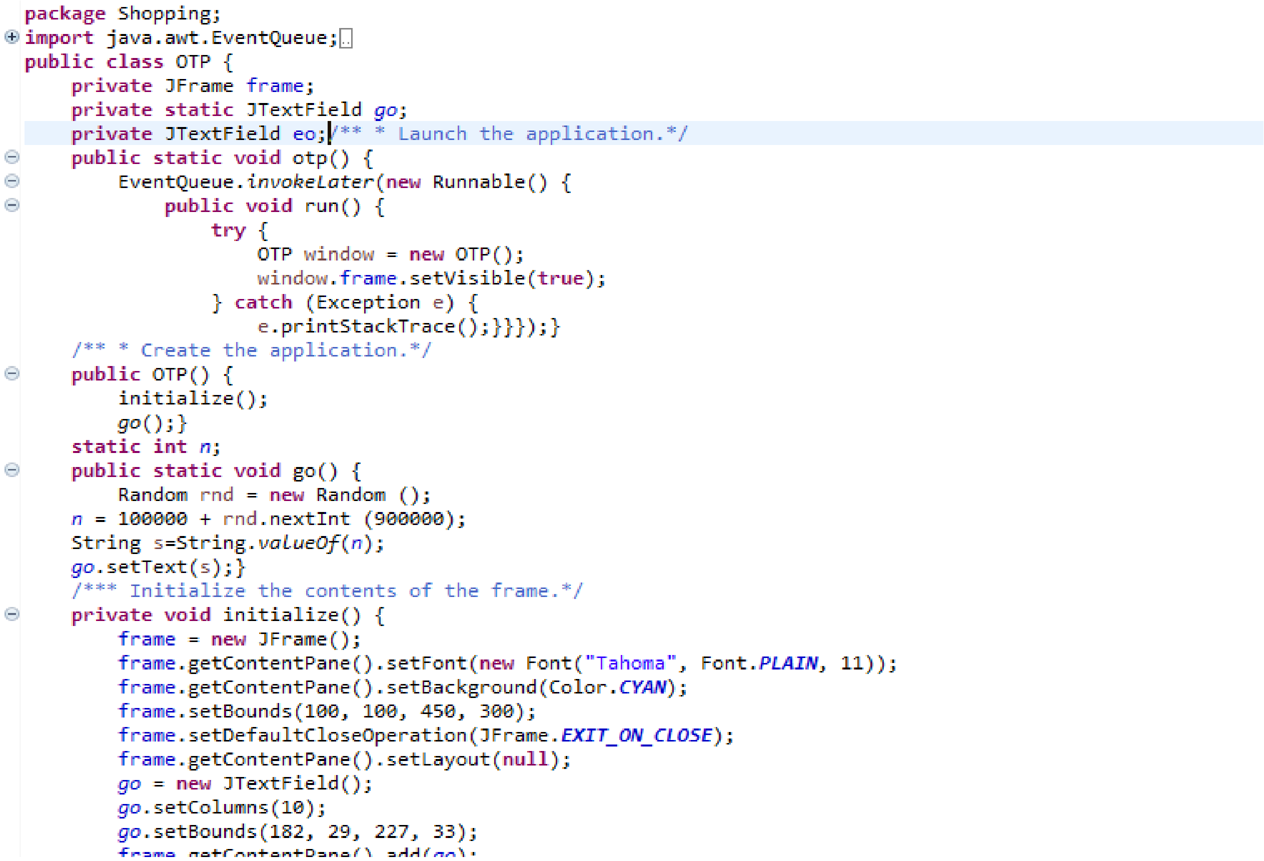


```
    public void actionPerformed(ActionEvent e) {
        String pi;
        pi = txtproids.getText();
        try {String q = "delete from ele2 where eid = ?";
            PreparedStatement pstmt = con.prepareStatement(q);
            pstmt.setString(1,pi);
            pstmt.executeUpdate();
            String p = "SET @num := 0;";
            String p3 = "UPDATE ele2 SET eid = @num := (@num+1);";
            String p4 = "ALTER TABLE ele2 AUTO_INCREMENT = 1;";
            java.sql.Statement stmt = con.createStatement();
            stmt.execute(p);
            stmt.execute(p3);
            stmt.execute(p4);
            JOptionPane.showMessageDialog(null,"record updated");
            txtproids.setText("");
            trypro.setText("");
            tryprocost.setText("");
            txtproids.requestFocus();
            t1();}
            catch(Exception ei) {
                ei.printStackTrace();}} });
btnDelete.setBounds(119, 153, 89, 28);
panel.add(btnDelete);
JPanel panel_2 = new JPanel();
panel_2.setLayout(null);
panel_2.setBackground(new Color(240, 248, 255));
panel_2.setBounds(20, 314, 277, 70);
frame.getContentPane().add(panel_2);
JLabel lblNewLabel_2 = new JLabel("SEARCH");
lblNewLabel_2.setBounds(10, 11, 66, 14);
panel_2.add(lblNewLabel_2);
JLabel lblNewLabel_1_2 = new JLabel("PRODUCT ID :");
lblNewLabel_1_2.setFont(new Font("Tahoma", Font.BOLD, 14));
lblNewLabel_1_2.setBounds(10, 31, 103, 28);
panel_2.add(lblNewLabel_1_2);
```

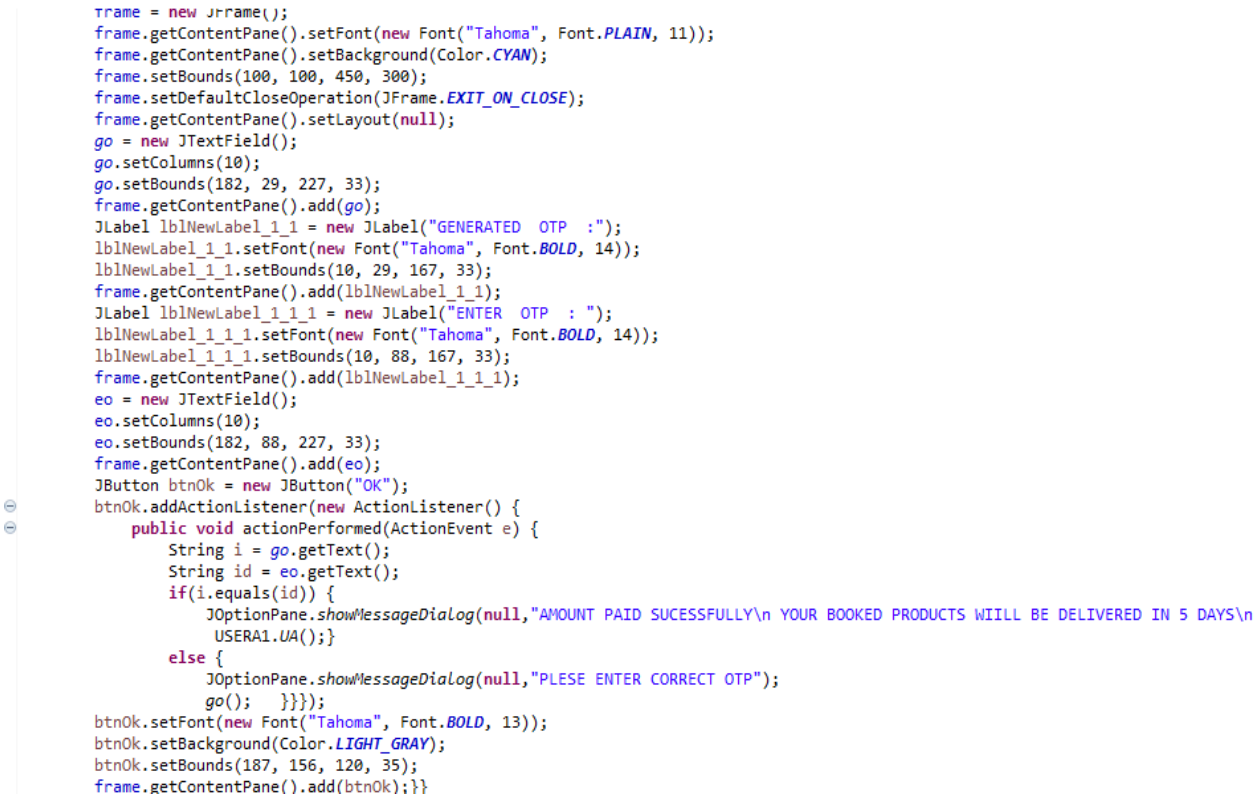


```
frame.getContentPane().add(panel_2);
JLabel lblNewLabel_2 = new JLabel("SEARCH");
lblNewLabel_2.setBounds(10, 11, 66, 14);
panel_2.add(lblNewLabel_2);
JLabel lblNewLabel_1_2 = new JLabel("PRODUCT ID :");
lblNewLabel_1_2.setFont(new Font("Tahoma", Font.BOLD, 14));
lblNewLabel_1_2.setBounds(10, 31, 103, 28);
panel_2.add(lblNewLabel_1_2);
```

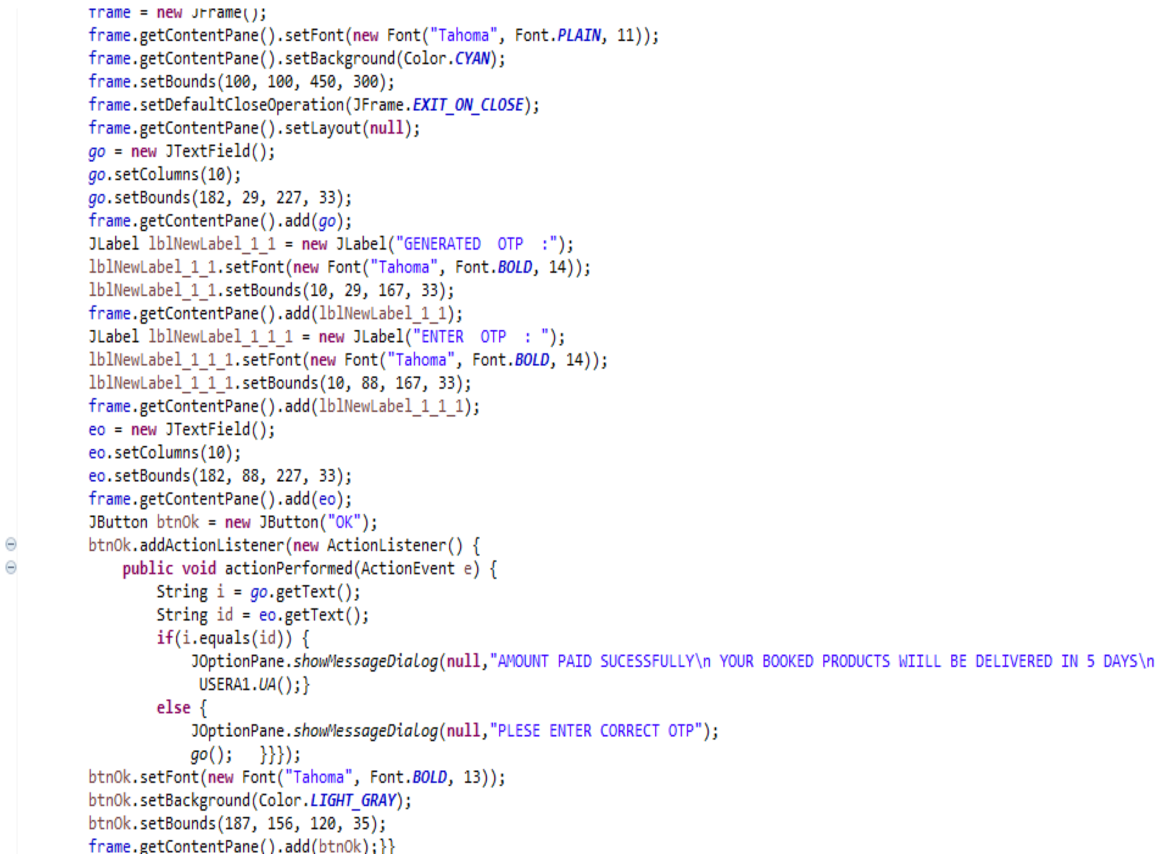
```
txtproids = new JTextField();
txtproids.addKeyListener(new KeyAdapter() {@Override
    public void keyReleased(KeyEvent e) {
        try {String id = txtproids.getText();
            PreparedStatement pst ;
            pst = con.prepareStatement("select eid,ename,ecost from ele2 where eid = ?");
            pst.setString(1,id);
            ResultSet set = pst.executeQuery();
            if(set.next() == true) {
                String eid = set.getString(1);
                String ename = set.getString(2);
                String ecost = set.getString(3);
                txtproids.setText(eid);
                trypro.setText(ename);
                tryprocost.setText(ecost);  }
            else {
                txtproids.setText("");
                trypro.setText("");
                tryprocost.setText("");}}
    catch(Exception ed) {}}});
txtproids.setColumns(10);
txtproids.setBounds(129, 37, 138, 20);
panel_2.add(txtproids);
JScrollPane scrollPane = new JScrollPane();
scrollPane.setBounds(287, 48, 371, 239);
frame.getContentPane().add(scrollPane);
table = new JTable();
scrollPane.setViewportView(table);}}
```



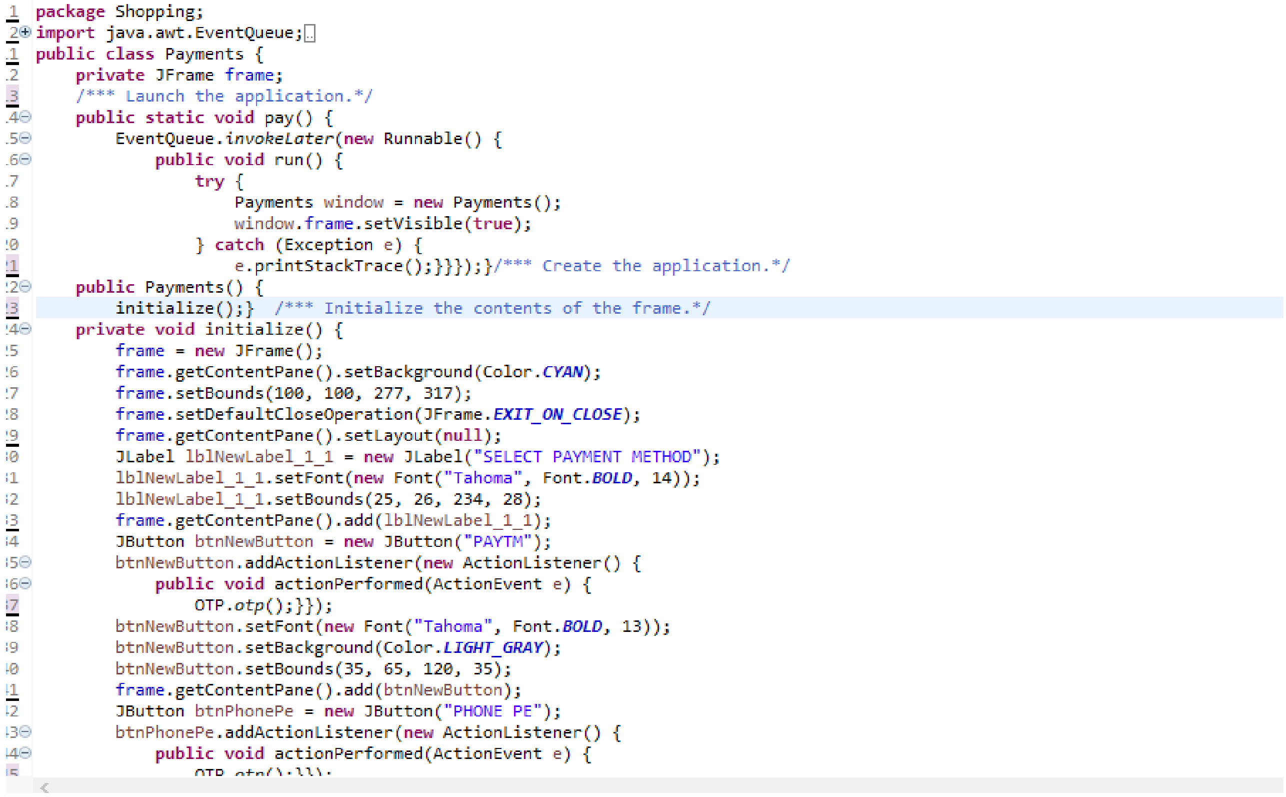
```
package Shopping;
import java.awt.EventQueue;
public class OTP {
    private JFrame frame;
    private static JTextField go;
    private JTextField eo;/** * Launch the application.*/
    public static void otp() {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    OTP window = new OTP();
                    window.frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();}}});}
    /** * Create the application.*/
    public OTP() {
        initialize();
        go();}
    static int n;
    public static void go() {
        Random rnd = new Random ();
    n = 100000 + rnd.nextInt (900000);
    String s=String.valueOf(n);
    go.setText(s);}
    /*** Initialize the contents of the frame.*/
    private void initialize() {
        frame = new JFrame();
        frame.getContentPane().setFont(new Font("Tahoma", Font.PLAIN, 11));
        frame.getContentPane().setBackground(Color.CYAN);
        frame.setBounds(100, 100, 450, 300);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.getContentPane().setLayout(null);
        go = new JTextField();
        go.setColumns(10);
        go.setBounds(182, 29, 227, 33);
        frame.getContentPane().add(go);
```



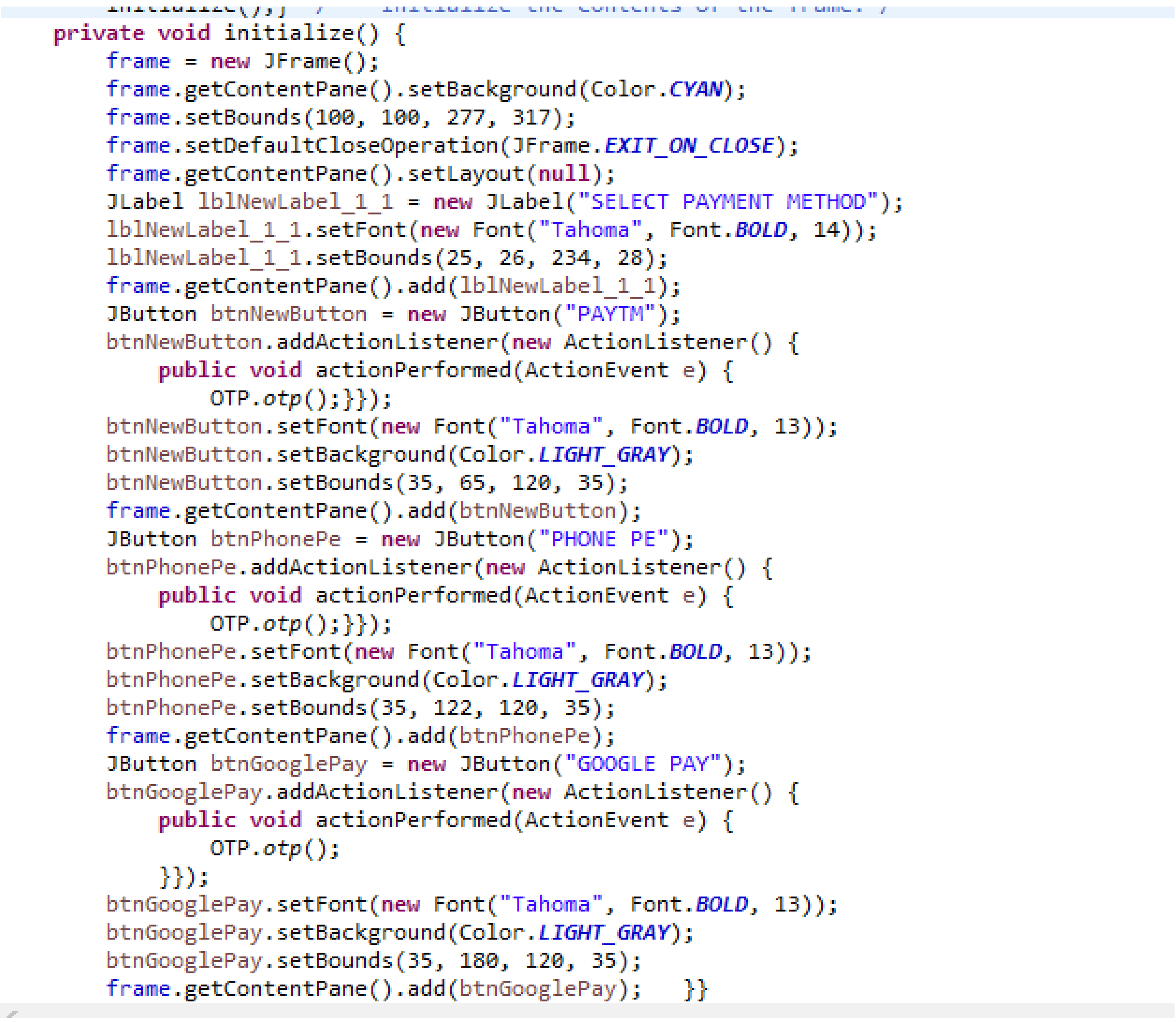
```
frame = new JFrame();
frame.getContentPane().setFont(new Font("Tahoma", Font.PLAIN, 11));
frame.getContentPane().setBackground(Color.CYAN);
frame.setBounds(100, 100, 450, 300);
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
frame.getContentPane().setLayout(null);
go = new JTextField();
go.setColumns(10);
go.setBounds(182, 29, 227, 33);
frame.getContentPane().add(go);
JLabel lblNewLabel_1_1 = new JLabel("GENERATED  OTP  :");
lblNewLabel_1_1.setFont(new Font("Tahoma", Font.BOLD, 14));
lblNewLabel_1_1.setBounds(10, 29, 167, 33);
frame.getContentPane().add(lblNewLabel_1_1);
JLabel lblNewLabel_1_1_1 = new JLabel("ENTER  OTP  : ");
lblNewLabel_1_1_1.setFont(new Font("Tahoma", Font.BOLD, 14));
lblNewLabel_1_1_1.setBounds(10, 88, 167, 33);
frame.getContentPane().add(lblNewLabel_1_1_1);
eo = new JTextField();
eo.setColumns(10);
eo.setBounds(182, 88, 227, 33);
frame.getContentPane().add(eo);
JButton btnOk = new JButton("OK");
btnOk.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String i = go.getText();
        String id = eo.getText();
        if(i.equals(id)) {
            JOptionPane.showMessageDialog(null,"AMOUNT PAID SUCESSFULLY\n YOUR BOOKED PRODUCTS WIILL BE DELIVERED IN 5 DAYS\n
             USERA1.UA();}
        else {
            JOptionPane.showMessageDialog(null,"PLESE ENTER CORRECT OTP");
            go();   }}});
btnOk.setFont(new Font("Tahoma", Font.BOLD, 13));
btnOk.setBackground(Color.LIGHT_GRAY);
btnOk.setBounds(187, 156, 120, 35);
frame.getContentPane().add(btnOk);}}
```



```
        frame = new JFrame();
        frame.getContentPane().setFont(new Font("Tahoma", Font.PLAIN, 11));
        frame.getContentPane().setBackground(Color.CYAN);
        frame.setBounds(100, 100, 450, 300);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.getContentPane().setLayout(null);
        go = new JTextField();
        go.setColumns(10);
        go.setBounds(182, 29, 227, 33);
        frame.getContentPane().add(go);
        JLabel lblNewLabel_1_1 = new JLabel("GENERATED  OTP  :");
        lblNewLabel_1_1.setFont(new Font("Tahoma", Font.BOLD, 14));
        lblNewLabel_1_1.setBounds(10, 29, 167, 33);
        frame.getContentPane().add(lblNewLabel_1_1);
        JLabel lblNewLabel_1_1_1 = new JLabel("ENTER  OTP  : ");
        lblNewLabel_1_1_1.setFont(new Font("Tahoma", Font.BOLD, 14));
        lblNewLabel_1_1_1.setBounds(10, 88, 167, 33);
        frame.getContentPane().add(lblNewLabel_1_1_1);
        eo = new JTextField();
        eo.setColumns(10);
        eo.setBounds(182, 88, 227, 33);
        frame.getContentPane().add(eo);
        JButton btnOk = new JButton("OK");
        btnOk.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                String i = go.getText();
                String id = eo.getText();
                if(i.equals(id)) {
                    JOptionPane.showMessageDialog(null,"AMOUNT PAID SUCESSFULLY\n YOUR BOOKED PRODUCTS WILLL BE DELIVERED IN 5 DAYS\n
                     USERA1.UA();}
                else {
                    JOptionPane.showMessageDialog(null,"PLESE ENTER CORRECT OTP");
                    go();   }}});
        btnOk.setFont(new Font("Tahoma", Font.BOLD, 13));
        btnOk.setBackground(Color.LIGHT_GRAY);
        btnOk.setBounds(187, 156, 120, 35);
        frame.getContentPane().add(btnOk);}}
```



```
package Shopping;
import java.awt.EventQueue;
public class Payments {
    private JFrame frame;
    /*** Launch the application.*/
    public static void pay() {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Payments window = new Payments();
                    window.frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();}}});}/*** Create the application.*/
    public Payments() {
        initialize();}  /*** Initialize the contents of the frame.*/
    private void initialize() {
        frame = new JFrame();
        frame.getContentPane().setBackground(Color.CYAN);
        frame.setBounds(100, 100, 277, 317);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.getContentPane().setLayout(null);
        JLabel lblNewLabel_1_1 = new JLabel("SELECT PAYMENT METHOD");
        lblNewLabel_1_1.setFont(new Font("Tahoma", Font.BOLD, 14));
        lblNewLabel_1_1.setBounds(25, 26, 234, 28);
        frame.getContentPane().add(lblNewLabel_1_1);
        JButton btnNewButton = new JButton("PAYTM");
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                OTP.otp();}});
        btnNewButton.setFont(new Font("Tahoma", Font.BOLD, 13));
        btnNewButton.setBackground(Color.LIGHT_GRAY);
        btnNewButton.setBounds(35, 65, 120, 35);
        frame.getContentPane().add(btnNewButton);
        JButton btnPhonePe = new JButton("PHONE PE");
        btnPhonePe.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                OTP.otp();}});
```



```
        initialize();}  /*   Initialize the contents of the frame. */
    private void initialize() {
        frame = new JFrame();
        frame.getContentPane().setBackground(Color.CYAN);
        frame.setBounds(100, 100, 277, 317);
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.getContentPane().setLayout(null);
        JLabel lblNewLabel_1_1 = new JLabel("SELECT PAYMENT METHOD");
        lblNewLabel_1_1.setFont(new Font("Tahoma", Font.BOLD, 14));
        lblNewLabel_1_1.setBounds(25, 26, 234, 28);
        frame.getContentPane().add(lblNewLabel_1_1);
        JButton btnNewButton = new JButton("PAYTM");
        btnNewButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                OTP.otp();}});
        btnNewButton.setFont(new Font("Tahoma", Font.BOLD, 13));
        btnNewButton.setBackground(Color.LIGHT_GRAY);
        btnNewButton.setBounds(35, 65, 120, 35);
        frame.getContentPane().add(btnNewButton);
        JButton btnPhonePe = new JButton("PHONE PE");
        btnPhonePe.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                OTP.otp();}});
        btnPhonePe.setFont(new Font("Tahoma", Font.BOLD, 13));
        btnPhonePe.setBackground(Color.LIGHT_GRAY);
        btnPhonePe.setBounds(35, 122, 120, 35);
        frame.getContentPane().add(btnPhonePe);
        JButton btnGooglePay = new JButton("GOOGLE PAY");
        btnGooglePay.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                OTP.otp();
            }});
        btnGooglePay.setFont(new Font("Tahoma", Font.BOLD, 13));
        btnGooglePay.setBackground(Color.LIGHT_GRAY);
        btnGooglePay.setBounds(35, 180, 120, 35);
        frame.getContentPane().add(btnGooglePay);    }}
```

CHAPTER 5

RESULTS

Fig 5.1: shows us the list of product catogryes and we should enter the catogry which has to be shown.

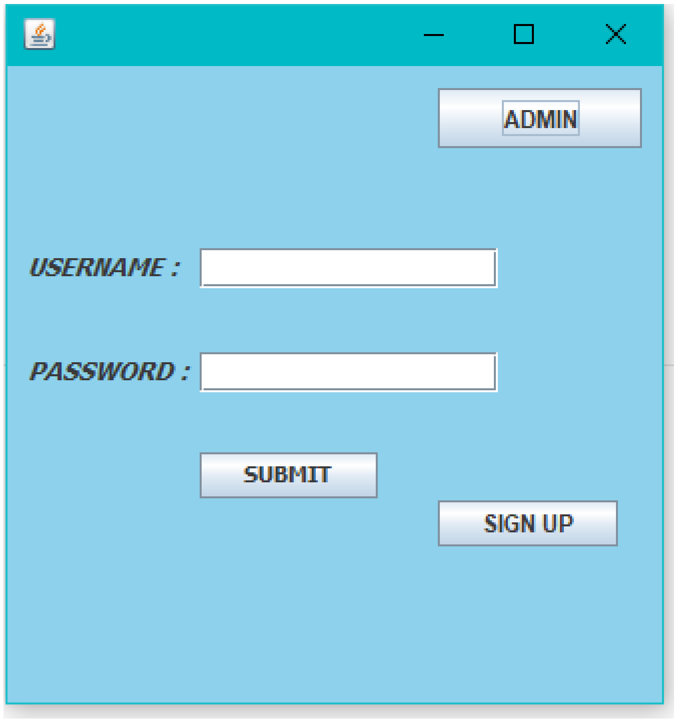
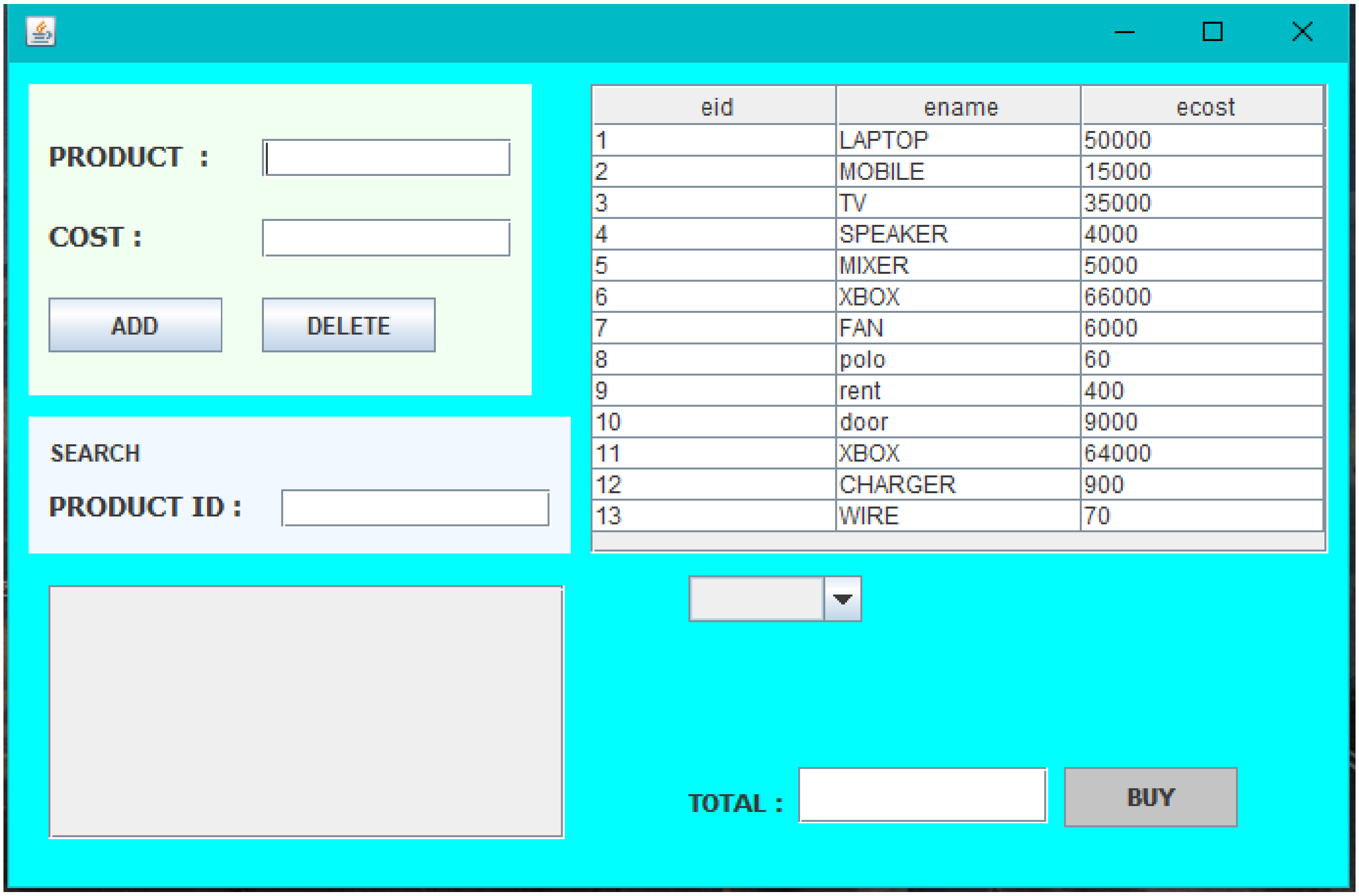
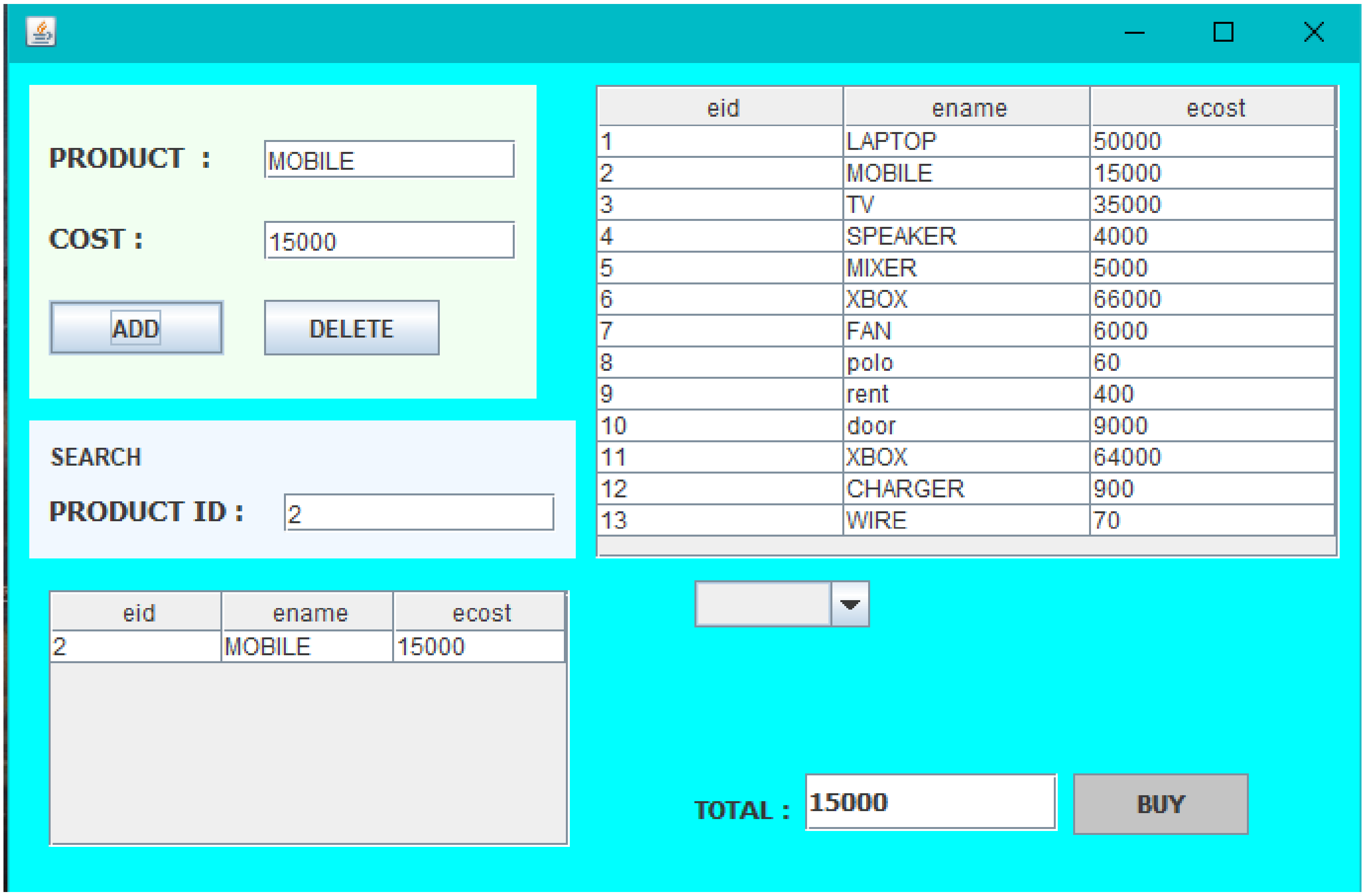


Fig 5.2 : 2 after entering the products to be booked we will enter into payment process 3 payment options are provided.



eid	ename	ecost
1	LAPTOP	50000
2	MOBILE	15000
3	TV	35000
4	SPEAKER	4000
5	MIXER	5000
6	XBOX	66000
7	FAN	6000
8	polo	60
9	rent	400
10	door	9000
11	XBOX	64000
12	CHARGER	900
13	WIRE	70

Fig 5.3: After choosing the payment option. We have to enter the amount to be paid. After paying the amount you should enter the otp which has displayed on the screen.



PRODUCT : MOBILE

COST : 15000

SEARCH

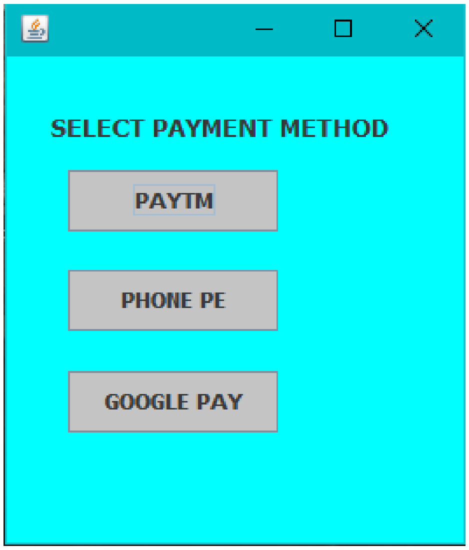
PRODUCT ID : 2

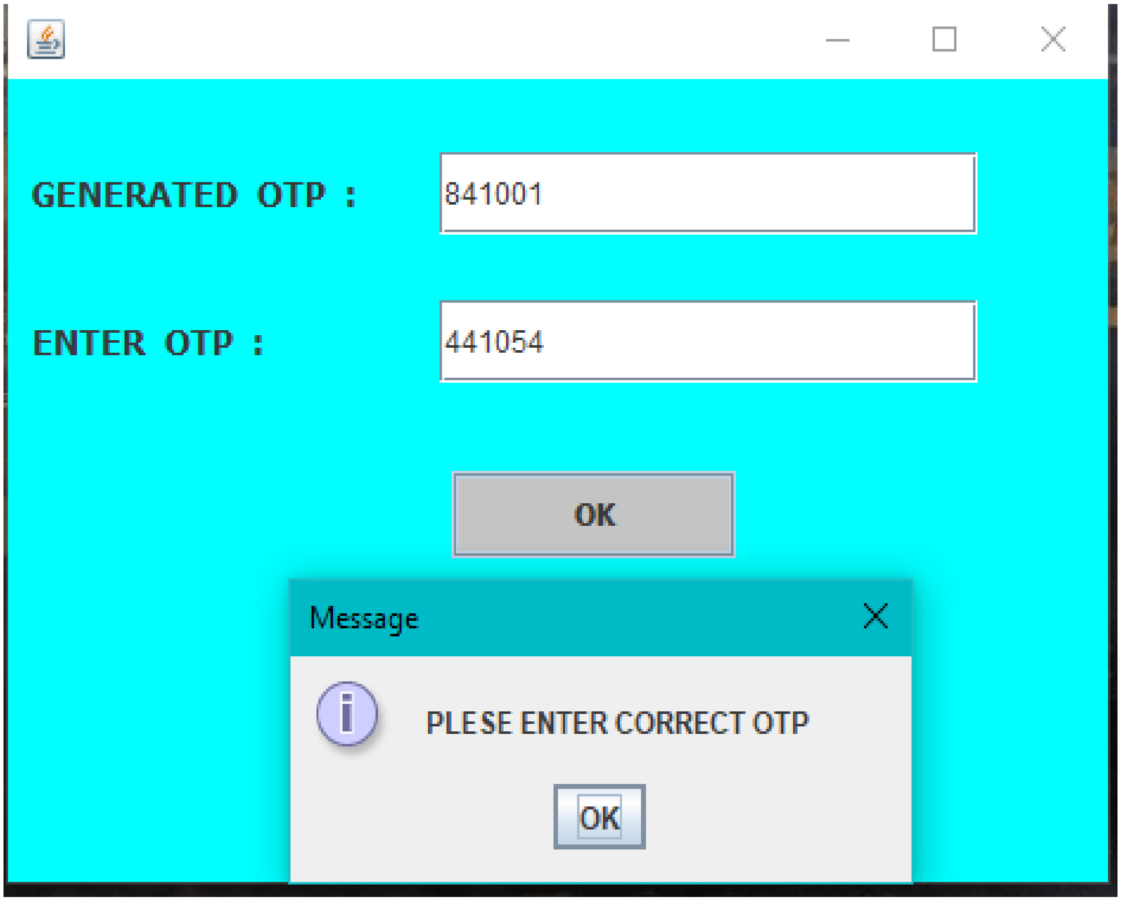
eid	ename	ecost
1	LAPTOP	50000
2	MOBILE	15000
3	TV	35000
4	SPEAKER	4000
5	MIXER	5000
6	XBOX	66000
7	FAN	6000
8	polo	60
9	rent	400
10	door	9000
11	XBOX	64000
12	CHARGER	900
13	WIRE	70

eid	ename	ecost
2	MOBILE	15000

TOTAL : 15000

Fig 5.4: we can see that after entering wrong otp .It shows WRONG OTP!!!. Then a new otp will be generated ..



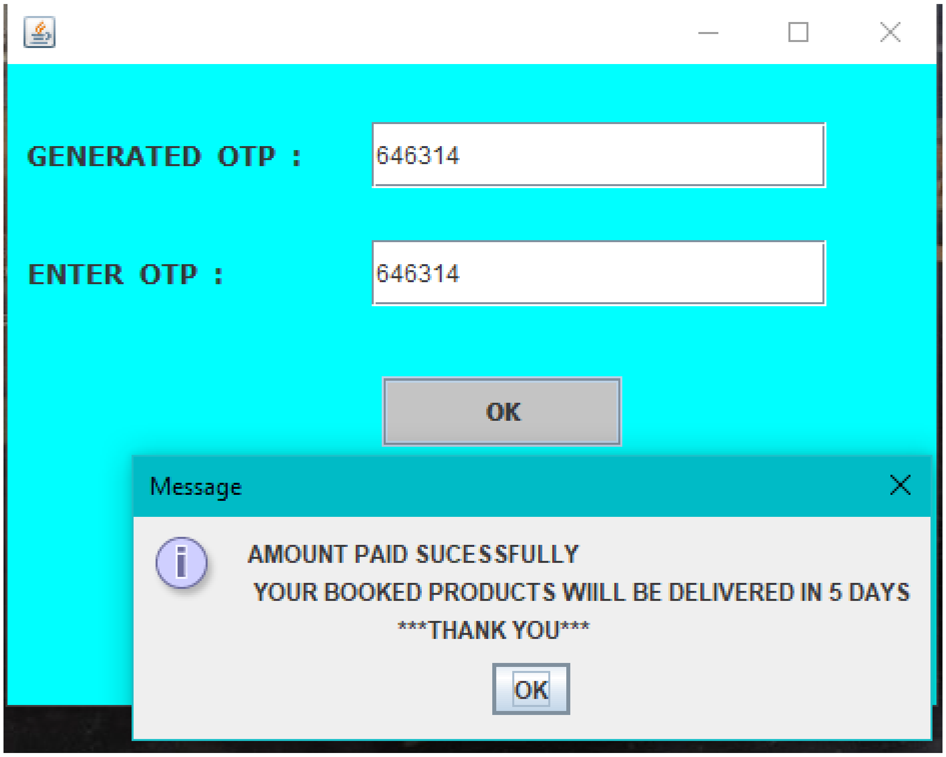

GENERATED OTP : 841001

ENTER OTP : 441054

Message

PLESE ENTER CORRECT OTP

Fig 5.5: After entering correct otp . products are successfully booked.



GENERATED OTP : 646314

ENTER OTP : 646314

Message

AMOUNT PAID SUCESSFULLY
YOUR BOOKED PRODUCTS WIILL BE DELIVERED IN 5 DAYS
THANK YOU

CONCLUSION

online shopping is one of the basic java application in java created using different OOPS concepts available.

online shopping has the basic options that is needed

1. Customer details
2. Customers age
3. No of days it will take to reach them.
4. Type of product catogries and its reqirements and customers payment method

This project is done successfully and also executed successfully without any errors In this project first customer has to enter their name and their age. they allowed for further process and they can book the products after that the customer needs to select the product after selecting the products then car price will be displayed on the screen and the payment options will be displayed and customer needs to select the option like google pay ,phonepe , paytm etc And after completion of payment OTP will be displayed and the customer need to enter the OTP then booking will complete.

REFERENCES

1. Herbert Schildt-Java_ The Complete Reference-McGraw-Hill Education (2017)
2. Head First Java: A Brain-Friendly Guide, 2nd Edition - Kathy Sierra, Bert Bates(2003)
3. PROGRAMMING WITH JAVA- PRIMER A
4. https://www.w3schools.com/java/java_oop.asp
5. https://www.javatpoint.com/java-tutorial

These are the books and the online links that I have referred to complete my mini project.