Appendix

Note: The following conversations were communicated in another language, hence, translated and comprehensible transcripts of our conversation has been provided below:

A.1: Initial conversation with client (phone call transcript) – 29th October, 2019

Client: Thanks for taking out your time for this, my managers and I have been having a tough time since long, and the storage space is being affected as well. I believe this storage space can be used for other useful purposes such as storing other products, accessories, etc.

Me: That's alright, I'm very excited to work on the solution to your problem!

Client: Well, that's great. Umm, where do I start? Okay, so my managers have been complaining to me about the manual system of generating invoices for the clients who've bought services or products from our salon.

Me: Oh, must be hard for them to manage everything right?

Client: Yes, definitely, let me tell you about the current system we're using. So basically, whenever a client uses a service or a product, an invoice using the invoice book is made. A copy for our records, and another copy for the client is made. It has become extremely tough to keep a record of all the invoices as each day, 100's of clients visit our salon, and if a specific invoice data is required, it becomes tough to find it in the storage due to how many are there.

Me: Yeah, I agree with you.

Client: Yes, so I was wondering if it would be possible for you to develop a computer program, which would manage all the invoices, and in addition to this, generate reports for how well our salon is doing as well, this is because the managers have not been able to analyse how our salon is doing in a specific time period.

Me: Oh, okay. So, you haven't been able to analyse how your salon has been doing?

Client: No, that's not the case. We have been able to do that, however, it's not very reliable as the managers may have made minor mistakes while making the invoice, and some invoice copies may also have been lost or unconsidered while analysing the salon's performance. Also, I would prefer a PDF format of the report and the invoice as it is very convenient and easy to refer to.

Me: Umm, sounds tough, but I think I'll manage. I've made a note of all your problems. Let me repeat whatever I've noted down. *{reading my notes}*

Client: Yes, perfect. Also, one more important thing, please make sure the program isn't too tough to use. Make sure it is extremely easy to use.

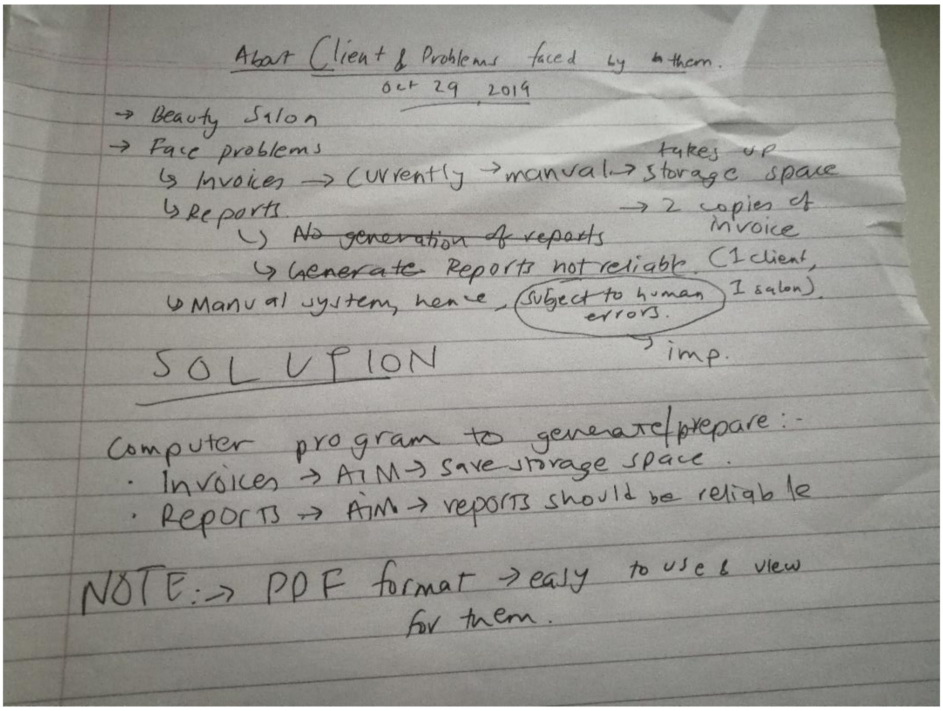
Me: Yes, for sure, I will keep that in mind while making the program. So, I will think of a success criterion and get back to you, okay?

Client: Sure, thanks a ton for this! I appreciate your help.

Me: No problem. Looking forward to making a useful program for you! Another thing, I will be using the database MySQL. There are several advantages of MySQL such as (refer to Criterion A Rationale for the Solution).

Client: Oh, wow, sounds pretty good to me. This database system sounds reliable, so I guess you should go for it.

Me: Okay



About Client & Problems faced by ~~to~~ them.

Oct 29 2019

→ Beauty Salon

→ Face problems

↳ Invoices → Currently → manual → takes up storage space

→ 2 copies of invoice

↳ Reports.

↳ ~~No generation of reports~~

↳ ~~Generate~~ Reports not reliable. (1 client, 1 salon).

↳ Manual system, hence, subject to human errors. → imp.

SOLUTION

Computer program to generate/prepare :-

· Invoices → AIM → Save storage space.

· Reports → AIM → reports should be reliable

NOTE:→ PDF format → easy to use & view for them.

An image of the notes I took during our phone call

A.2: Conversation 2 with client (face to face meet transcript) – 8th November, 2019

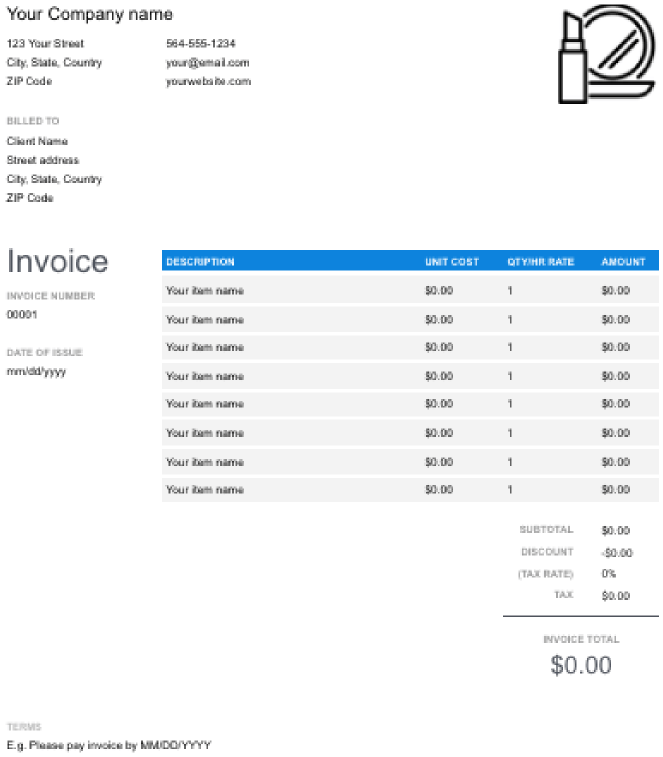
Client: Before you show me the criterion you've prepared, I found other alternative options, such as directly buying a software which would do the work for me and spare you all the time and efforts you'll have to put. This software was known as miosalon[1]. However, it looked extremely complicated and the main disadvantage, it could only be accessed through on internet. So, please make it as simple to use or user friendly as possible. I don't want my managers to complain about how tough the program is to use.

Me: Oh, okay, however, if you decide to use such an advanced program, and that too online, that's your call I guess, but I agree with your point, it becomes quite complicated and also, if you have internet issues, it can affect your salon's business. Okay, moving on, this is my success criterion. *{showing the success criteria from criterion A}*

Client: Looks good to me, I noticed you added a module for branch information, that's useful as well. Good suggestion. Make sure you follow these points thoroughly!

Me: Yes, for sure! I've made this list in accordance with my capability, so I'm sure I'll be able to do it.

Client: One more thing, regarding the PDF format of the invoice, this is a layout I found on google. It would help if the program generated an invoice in this format.



Your Company name

123 Your Street
City, State, Country
ZIP Code

564-555-1234
your@email.com
yourwebsite.com

BILLED TO
Client Name
Street address
City, State, Country
ZIP Code

Invoice

INVOICE NUMBER
00001

DATE OF ISSUE
mm/dd/yyyy

DESCRIPTION	UNIT COST	QTY/HR RATE	AMOUNT
Your item name	$0.00	1	$0.00
Your item name	$0.00	1	$0.00
Your item name	$0.00	1	$0.00
Your item name	$0.00	1	$0.00
Your item name	$0.00	1	$0.00
Your item name	$0.00	1	$0.00
Your item name	$0.00	1	$0.00
Your item name	$0.00	1	$0.00

SUBTOTAL	$0.00
DISCOUNT	-$0.00
(TAX RATE)	0%
TAX	$0.00

INVOICE TOTAL
$0.00

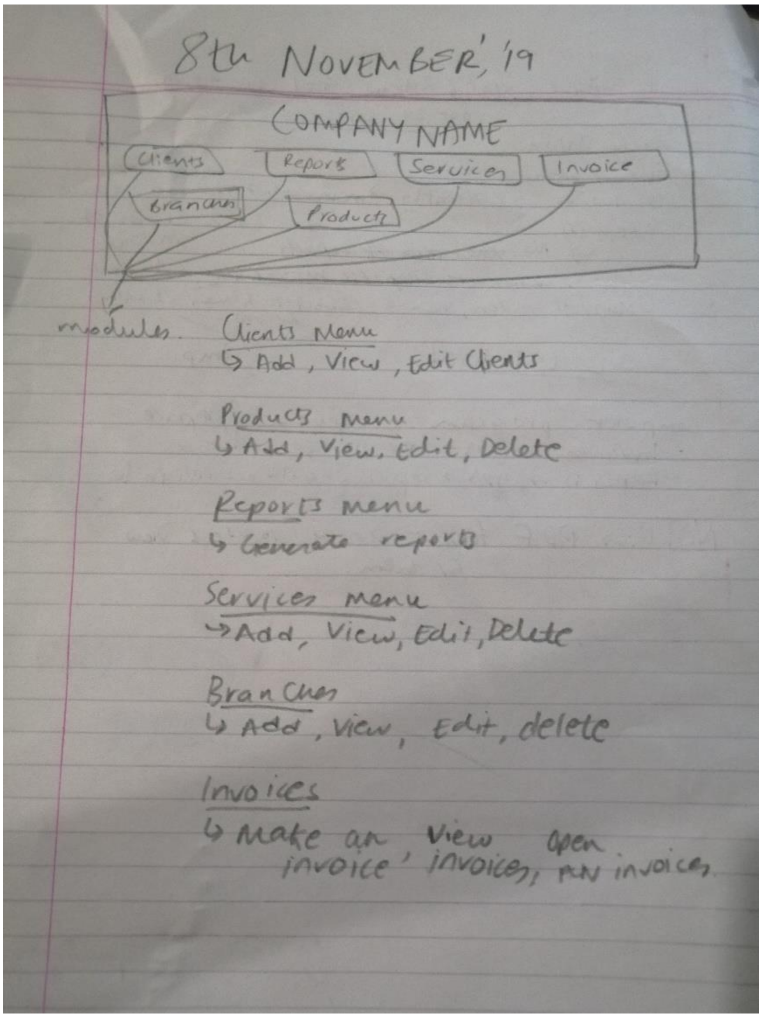
TERMS
E.g. Please pay invoice by MM/DD/YYYY

An image of how the PDF invoice should look, as suggested by the client

Me: Oh okay, this is a good template. I will try my best to follow this one.

[1] https://www.miosalon.com/?gclid=CjwKCAiA-P7xBRAvEiwAow-VaeejsJAXOkBmYj7ex97esfHGw-eQv3MdYXhynI31-iuAdJYZMwvgfxoCyhYQAvD_BwE#homeFeature

Client: Great! Also, for your reference, I've prepared a sample layout of the program and details of each module. It would be convenient if you could make something similar to this.


COMPANY NAME
Clients
Reports
Services
Invoice
Branches
Products

8th NOVEMBER, '19

modules. Clients Menu
↳ Add, View, Edit Clients

Products Menu
↳ Add, View, Edit, Delete

Reports menu
↳ Generate reports

Services menu
→ Add, View, Edit, Delete

Branches
↳ Add, View, Edit, delete

Invoices
↳ Make an invoice, View open invoices, All invoices

An image of the layout of the program

Client: Make sure you don't lose this paper and keep it safely with you. It also has the important notes you took down as well, that's why.

Me: Yes, I will not lose it for sure.

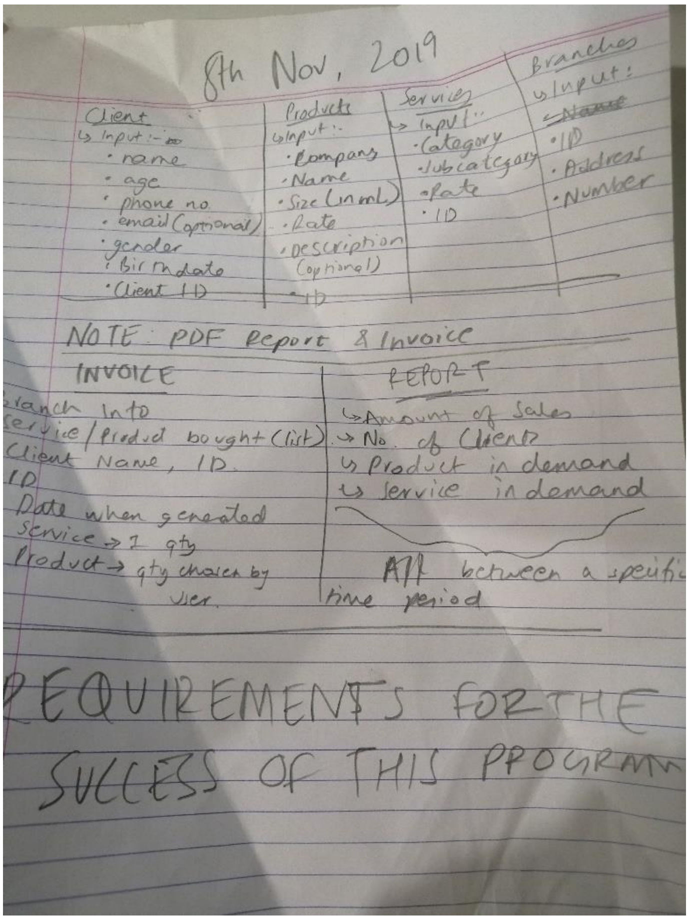
Client: Good, also, I think you will also require the information that should be inputted right?

Me: Oh right, I totally forgot about that part. I do need to know what all has to be entered, in detail.

Client: Okay, I will write whatever is needed in detail.

Me: Thanks a lot! This will help me to develop the program in the exact way you want!

Client: Great, I'm glad to hear that! Okay so take a look at this information.



8th Nov, 2019

Client
↳ Input :-
- name
- age
- phone no.
- email (optional)
- gender
- Birthdate
- Client ID

Products
↳ Input :-
- Company
- Name
- Size (in mL)
- Rate
- Description (optional)
- ID

Services
↳ Input :-
- Category
- Subcategory
- Rate
- ID

Branches
↳ Input :
~~Name~~
- ID
- Address
- Number

NOTE: PDF Report & Invoice

INVOICE

Branch Info
Service/Product bought (list)
Client Name, ID.
ID
Date when generated
Service → 1 qty
Product → qty chosen by user.

REPORT

↳ Amount of Sales
→ No. of Clients
↳ Product in demand
↳ Service in demand

All between a specific time period

REQUIREMENTS FOR THE SUCCESS OF THIS PROGRAM

An image of the information to be entered by the manager

Me: Wow, this clarifies a lot of doubts. This is very detailed, and I know exactly what I have to do now. Thank you

Client: Alright, excellent. Now, you can start working on it. When can I expect this program by?

Me: Umm, let me think. So, I have my exams coming up soon, so I will be balancing this with other external work. So latest by the end of December or early January?

Client: Hmm, sounds pretty good to me.

A.3: Conversation 3 with client (face to face meet transcript) – 28th December, 2019

Showing the design of my program to the client and client uses the program.

Client: This program is extremely easy to use! You've exceeded my expectations for sure! This is exactly how I wanted my program to be. You've also added validation and verification of data. Good. Okay, so, you're on the verge of completion, continue your work!

A.3.1: *More discussions regarding the solution and more feedback was given by client regarding the program*

A.4: Conversation 4 with client (face to face meet transcript) – 30th December, 2019

Me: Here is the final product. Please contact me if any changes are to be made.

Client: Well, it looks extremely good, I don't really think any changes to the layout or the design should be made. However, if I find any errors in the program, which is highly unlikely, I will surely contact you.

Me: Okay, great. I had a lot of fun while developing this program, along with having a hard time as well of course.

Client: That's great, I'm sure you got to learn a lot from this project too.

Me: Yes, surely. I learnt quite a lot from this whole experience.

Client: Another thing, now that you've successfully made this program. I would like you to consider other features to this program as well.

Me: Umm okay. Can you give me examples of such features?

Client: Okay, so while viewing the list of clients, products, etc., even though the rows have been sorted alphabetically, which makes it easier to find the client, I would want to search from a search bar, similar to that of search engines like google, which makes it much easier to look for specific data.

Me: Oh right, that's a great idea. Any other ideas you have in mind?

Client: Yes, I do have another idea, like adding an appointment system as if a client calls the salon and wants to book for an appointment, the needful can be done easily using an appointment system.

Me: Okay, this is a great idea as well. I will definitely try my best to add these features in the program in the future

Client: If you do, that would be even better. Nevertheless, the current program also works extremely well. Once again, thanks a lot for this. It will surely help our managers and I'm hoping that I don't get any more complaints regarding this issue.

Me: Yes, I hope so too. I'm pretty sure my program will not have errors.

Java Code divided by each class

1. AddBranch.java

```java
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.KeyAdapter;
import java.awt.event.KeyEvent;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.Random;

import javax.swing.JButton;
import javax.swing.JFormattedTextField;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JTextField;
import javax.swing.border.EmptyBorder;

public class AddBranch extends JFrame {

      private JPanel contentPane;
      private final JPanel panel = new JPanel();
      private final JPanel panel_1 = new JPanel();
      JLabel lblAddress = new JLabel("Address:");
      JLabel lblContactNumber = new JLabel("Contact Number:");
      private JTextField branchID;
      JFormattedTextField branchAddress;
      JFormattedTextField branchContact;
      char vchar;
      /**
       * Launch the application.
       */
      public static void main(String[] args) {
            EventQueue.invokeLater(new Runnable() {
                  public void run() {
                        try {
                              AddBranch frame = new AddBranch();
                              frame.setVisible(true);
                        } catch (Exception e) {
                              e.printStackTrace();
                        }
                  }
            });
      }

      /**
       * Create the frame.
       */

      String branch = "BRANCH";
```

```
PreparedStatement pst;
Random r = new Random();
ResultSet rs;
String squery;
String query;
String id;
Connection con;
public AddBranch() {
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setBounds(100, 300, 877, 226);
    contentPane = new JPanel();
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
    setContentPane(contentPane);
    contentPane.setLayout(null);
    panel.setBackground(new Color(240, 255, 255));
    panel.setBounds(0, 0, 863, 36);
    contentPane.add(panel);
    panel.setLayout(null);

    JLabel lblNewLabel = new JLabel("Add a Branch\r\n");
    lblNewLabel.setForeground(new Color(0, 124, 124));
    lblNewLabel.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
    lblNewLabel.setBounds(0, 0, 1108, 36);
    panel.add(lblNewLabel);
    panel_1.setBackground(Color.DARK_GRAY);
    panel_1.setBounds(0, 31, 863, 158);

    JLabel lblNewLabel_3 = new JLabel("");
    lblNewLabel_3.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
    lblNewLabel_3.setForeground(Color.RED);
    lblNewLabel_3.setBounds(445, 87, 176, 23);
    panel_1.add(lblNewLabel_3);

    contentPane.add(panel_1);
    panel_1.setLayout(null);

    JLabel lblNewLabel_2 = new JLabel("");
    lblNewLabel_2.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
    panel_1.add(lblNewLabel_2);

    branchID = new JTextField();
    branchID.setBounds(152, 23, 469, 23);
    branchID.setEditable(false);
    branchID.setBackground(Color.white);
    branchID.setForeground(Color.black);
    branchID.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
    panel_1.add(branchID);
    branchID.setColumns(10);

    JLabel lblNewLabel_1 = new JLabel("Branch ID:");
    lblNewLabel_1.setForeground(Color.WHITE);
    lblNewLabel_1.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
    lblNewLabel_1.setBounds(10, 21, 132, 23);
    panel_1.add(lblNewLabel_1);

    lblAddress.setForeground(Color.WHITE);
    lblAddress.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
    lblAddress.setBounds(10, 54, 132, 23);
    panel_1.add(lblAddress);
```

```
            lblContactNumber.setForeground(Color.WHITE);
            lblContactNumber.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
            lblContactNumber.setBounds(10, 87, 132, 23);
            panel_1.add(lblContactNumber);

            branchAddress = new JFormattedTextField();
            branchAddress.setBounds(152, 56, 698, 23);
            panel_1.add(branchAddress);

            branchContact = new JFormattedTextField();
            branchContact.addKeyListener(new KeyAdapter() {
                @Override
                public void keyTyped(KeyEvent e) {
                    vchar = e.getKeyChar();
                    if ((!(Character.isDigit(vchar)) ||  (vchar ==
KeyEvent.VK_BACK_SPACE) ||  (vchar == KeyEvent.VK_DELETE))) {
                        lblNewLabel_3.setText("Invalid Entry!");
                        e.consume();
                    }
                    else {
                        lblNewLabel_3.setText("");
                    }
                }
            });
            branchContact.setBounds(152, 87, 283, 23);
            panel_1.add(branchContact);

            id = branch + (System.currentTimeMillis());
            branchID.setText(id);

            JButton button = new JButton("Add a new Branch");
            button.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    try {
                        con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                        query = "INSERT INTO `csiatables`.`branchlist`
(`BranchID`, `Address`, `ContactNumber`) VALUES (?, ?, ?)";
                        pst = con.prepareStatement(query);
                        pst.setString(1, id);
                        if (branchAddress.getText().isEmpty() == false &&
branchContact.getText().isEmpty() == false) {
                            if (branchContact.getText().length()==10)
{
                                pst.setString(2,
branchAddress.getText());
                                pst.setString(3,
branchContact.getText());
                            }
                            else {
                                JOptionPane.showMessageDialog(null,
"Contact Number should be 10 digits long");
                            }
                        }
                        pst.executeUpdate();
                        JOptionPane.showMessageDialog(null, "Added branch
successfully");
```

```
                    id = branch + (System.currentTimeMillis());
                    branchID.setText(id);
                    branchAddress.setText("");
                    branchContact.setText("");
                    }
                catch(Exception e1) {
                    if (branchAddress.getText().toString().length()
== 0 || branchContact.getText().toString().length() == 0) {
                        JOptionPane.showMessageDialog(null, "One
or more fields have been left blank");
                    }
                }
            }
        });
        button.setForeground(Color.BLACK);
        button.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button.setBackground(Color.WHITE);
        button.setBounds(152, 120, 167, 23);
        panel_1.add(button);

        JButton btnClearContents = new JButton("Clear Contents");
        btnClearContents.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                lblNewLabel_3.setText("");
                branchAddress.setText("");
                branchContact.setText("");
            }
        });
        btnClearContents.setForeground(Color.BLACK);
        btnClearContents.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnClearContents.setBackground(Color.WHITE);
        btnClearContents.setBounds(329, 120, 167, 23);
        panel_1.add(btnClearContents);

        JButton btnBack = new JButton("Back");
        btnBack.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                dispose();
            }
        });
        btnBack.setForeground(Color.BLACK);
        btnBack.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnBack.setBackground(Color.WHITE);
        btnBack.setBounds(506, 120, 167, 23);
        panel_1.add(btnBack);
    }
}
```

2. AddClient.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.KeyAdapter;
import java.awt.event.KeyEvent;
import java.sql.Connection;
import java.sql.DriverManager;
```

```java
import java.sql.PreparedStatement;
import java.text.SimpleDateFormat;

import javax.swing.ButtonGroup;
import javax.swing.JButton;
import javax.swing.JFormattedTextField;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JRadioButton;
import javax.swing.JTextField;
import javax.swing.border.EmptyBorder;

import com.toedter.calendar.JDateChooser;

public class AddClient extends JFrame {

	private JPanel contentPane;
	private JTextField clientID;
	JFormattedTextField clientContact, clientName, clientEmail;
	JDateChooser bdayChoose;

	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {
				try {
					AddClient frame = new AddClient();
					frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}

	/**
	 * Create the frame.
	 */

	public void clearContents() {
		clientEmail.setText("");
		clientName.setText("");
		clientContact.setText("");
		bdayChoose.setDate(null);
	}
	String client = "CLIENT";
	String id, date;
	SimpleDateFormat sdf;
	String gender;
	char vchar;
	Connection con;
	String query;
	PreparedStatement pst;
	public AddClient(){
		setBounds(100, 300, 706, 286);
```

```
contentPane = new JPanel();
contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
setContentPane(contentPane);
contentPane.setLayout(null);

JPanel panel = new JPanel();
panel.setLayout(null);
panel.setBackground(new Color(240, 255, 255));
panel.setBounds(0, 0, 863, 36);
contentPane.add(panel);

JLabel lblAddANew = new JLabel("Add a new Client\r\n");
lblAddANew.setForeground(new Color(0, 124, 124));
lblAddANew.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
lblAddANew.setBounds(0, 0, 692, 26);
panel.add(lblAddANew);

JPanel panel_1 = new JPanel();
panel_1.setLayout(null);
panel_1.setBackground(Color.DARK_GRAY);
panel_1.setBounds(0, 31, 692, 218);
contentPane.add(panel_1);

JLabel label_2 = new JLabel("");
label_2.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
label_2.setBounds(0, 0, 0, 0);
panel_1.add(label_2);

clientID = new JTextField();
clientID.setForeground(Color.BLACK);
clientID.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
clientID.setEditable(false);
clientID.setColumns(10);
clientID.setBackground(Color.WHITE);
clientID.setBounds(152, 22, 283, 23);
panel_1.add(clientID);

JRadioButton rdbtnMale = new JRadioButton("Male");
rdbtnMale.setForeground(Color.WHITE);
rdbtnMale.setBackground(Color.DARK_GRAY);
rdbtnMale.setFont(new Font("Bahnschrift", Font.BOLD, 12));
rdbtnMale.setBounds(152, 154, 51, 21);
panel_1.add(rdbtnMale);

JRadioButton rdbtnFemale = new JRadioButton("Female");
rdbtnFemale.setForeground(Color.WHITE);
rdbtnFemale.setFont(new Font("Bahnschrift", Font.BOLD, 12));
rdbtnFemale.setBackground(Color.DARK_GRAY);
rdbtnFemale.setBounds(205, 154, 80, 21);
panel_1.add(rdbtnFemale);

JLabel lblClientId = new JLabel("Client ID:");
lblClientId.setForeground(Color.WHITE);
lblClientId.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
lblClientId.setBounds(10, 23, 132, 23);
panel_1.add(lblClientId);

JLabel lblMobileNumber = new JLabel("Mobile Number:");
lblMobileNumber.setForeground(Color.WHITE);
```

```
        lblMobileNumber.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblMobileNumber.setBounds(10, 85, 132, 23);
        panel_1.add(lblMobileNumber);

        JLabel lblNewLabel_3 = new JLabel("");
        lblNewLabel_3.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblNewLabel_3.setForeground(Color.RED);
        lblNewLabel_3.setBounds(445, 87, 176, 23);
        panel_1.add(lblNewLabel_3);

        clientContact = new JFormattedTextField();
        clientContact.addKeyListener(new KeyAdapter() {
            @Override
            public void keyTyped(KeyEvent e) {
                vchar = e.getKeyChar();
                if ((!(Character.isDigit(vchar)) ||  (vchar ==
KeyEvent.VK_BACK_SPACE) ||  (vchar == KeyEvent.VK_DELETE))) {
                    lblNewLabel_3.setText("Invalid Entry!");
                    e.consume();
                }
                else {
                    lblNewLabel_3.setText("");
                }
            }
        });
        clientContact.setBounds(152, 87, 283, 23);
        panel_1.add(clientContact);

        clientName = new JFormattedTextField();
        clientName.setBounds(152, 56, 283, 23);
        panel_1.add(clientName);

        clientEmail = new JFormattedTextField();
        clientEmail.setBounds(152, 120, 522, 23);
        panel_1.add(clientEmail);

        id = client + (System.currentTimeMillis()/1000);
        clientID.setText(id);

        bdayChoose = new JDateChooser();
        bdayChoose.setBounds(371, 151, 250, 23);
        bdayChoose.setDateFormatString("yyyy/MM/dd");
        panel_1.add(bdayChoose);

        JButton btnAddClient = new JButton("Add Client");
        btnAddClient.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                if (rdbtnMale.isSelected()==true)
                {gender = "M";}
                else if (rdbtnFemale.isSelected()==true)
                {gender = "F";}
                try {
                    con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                    query = "INSERT INTO `csiatables`.`clientlist`
(`ClientID`, `Name`, `MobileNumber`, `Email`, `Gender`, `Birthday`)"
                            + " VALUES (?, ?, ?, ?, ?, ?)";
                    pst = con.prepareStatement(query);
                    pst.setString(1, id);
                    if (clientName.getText().isEmpty() == false &&
clientContact.getText().isEmpty() == false) {
                        sdf = new SimpleDateFormat("yyyy/MM/dd");
                        date = sdf.format(bdayChoose.getDate());
                        pst.setString(2, clientName.getText());
                        pst.setString(3, clientContact.getText());
                        pst.setString(4,
clientEmail.getText().toLowerCase());
                        pst.setString(5, gender);
                        pst.setString(6, date);
                    }
                    pst.executeUpdate();
                    JOptionPane.showMessageDialog(null, "Added Client
Successfully");
                    id = client + (System.currentTimeMillis()/1000);
                    clientID.setText(id);
                    clearContents();
                }
                catch(Exception e1) {
                    JOptionPane.showMessageDialog(null, e1);
                }
            }
        });
        btnAddClient.setForeground(Color.BLACK);
        btnAddClient.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnAddClient.setBackground(Color.WHITE);
        btnAddClient.setBounds(152, 181, 167, 23);
        panel_1.add(btnAddClient);

        JButton button_1 = new JButton("Clear Contents");
        button_1.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                clearContents();
            }
        });
        button_1.setForeground(Color.BLACK);
        button_1.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_1.setBackground(Color.WHITE);
        button_1.setBounds(329, 181, 167, 23);
        panel_1.add(button_1);

        JButton button_2 = new JButton("Back");
        button_2.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                dispose();
            }
        });
        button_2.setForeground(Color.BLACK);
        button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_2.setBackground(Color.WHITE);
        button_2.setBounds(506, 181, 167, 23);
        panel_1.add(button_2);

        JLabel lblEmail = new JLabel("Email:");
        lblEmail.setForeground(Color.WHITE);
        lblEmail.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblEmail.setBounds(10, 116, 132, 23);
        panel_1.add(lblEmail);
        JLabel lblGender = new JLabel("Gender:");
        lblGender.setForeground(Color.WHITE);
        lblGender.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblGender.setBounds(10, 147, 132, 23);
        panel_1.add(lblGender);

        ButtonGroup bg = new ButtonGroup();
        bg.add(rdbtnMale);
        bg.add(rdbtnFemale);

        JLabel lblBirthdate = new JLabel("Birthdate:");
        lblBirthdate.setForeground(Color.WHITE);
        lblBirthdate.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblBirthdate.setBounds(291, 153, 85, 23);
        panel_1.add(lblBirthdate);

        JLabel lblName = new JLabel("Name");
        lblName.setForeground(Color.WHITE);
        lblName.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblName.setBounds(10, 54, 132, 23);
        panel_1.add(lblName);
    }
}
```

3. AddProduct.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.KeyAdapter;
import java.awt.event.KeyEvent;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;

import javax.swing.JButton;
import javax.swing.JFormattedTextField;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JTextArea;
import javax.swing.border.EmptyBorder;

public class AddProduct extends JFrame {

    private JPanel contentPane;
    JFormattedTextField company, prodname, size, rate;
    JTextArea description;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                    try {
                        AddProduct frame = new AddProduct();
                        frame.setVisible(true);
                    } catch (Exception e) {
                        e.printStackTrace();
                    }
            }
        });
    }

    /**
     * Create the frame.
     */

    PreparedStatement pst;
    String query, comp, pname, desc;
    double size1, rate1;
    Connection con;
    public void clearContents() {
        company.setText("");
        prodname.setText("");
        size.setText("");
        rate.setText("");
        description.setText("");
    }

    public AddProduct() {
        setBounds(100, 300, 626, 264);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setLayout(null);
        panel.setBackground(new Color(240, 255, 255));
        panel.setBounds(0, 0, 612, 28);
        contentPane.add(panel);

        JLabel lblAddANew = new JLabel("Add a new Product");
        lblAddANew.setForeground(new Color(0, 124, 124));
        lblAddANew.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
        lblAddANew.setBounds(0, 0, 692, 26);
        panel.add(lblAddANew);

        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBackground(Color.DARK_GRAY);
        panel_1.setBounds(0, 30, 612, 229);
        contentPane.add(panel_1);

        JLabel lblCompany = new JLabel("Company:");
        lblCompany.setForeground(Color.WHITE);
        lblCompany.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblCompany.setBounds(10, 8, 86, 23);
        panel_1.add(lblCompany);

        JLabel lblSizeinMl = new JLabel("Size (in ML):");
        lblSizeinMl.setForeground(Color.WHITE);
```

```
        lblSizeinMl.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblSizeinMl.setBounds(10, 72, 94, 23);
        panel_1.add(lblSizeinMl);

        JButton btnAddProduct = new JButton("Add Product");
        btnAddProduct.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                query = "INSERT INTO `csiatables`.`productlist`
(`Company`, `ProductName`, `ProductDescription`, `Size`, `Rate`) VALUES (?, ?, ?,
?, ?)";
                comp = company.getText();
                pname = prodname.getText();
                size1 = Double.parseDouble((size.getText()));
                rate1 = Double.parseDouble(rate.getText());
                desc = description.getText();
                try {
                    con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                    pst = con.prepareStatement(query);
                    if (comp.isEmpty() == false && pname.isEmpty() ==
false && size.getText().isEmpty() == false && rate.getText().isEmpty() == false) {
                        pst.setString(1, comp);
                        pst.setString(2, pname);
                        pst.setString(3, desc);
                        pst.setDouble(4, size1);
                        pst.setDouble(5, rate1);
                        pst.executeUpdate();
                        JOptionPane.showMessageDialog(null,
"Product added successfully!");
                        clearContents();
                    }
                    else {
                        JOptionPane.showMessageDialog(null, "One
or more fields are empty, please fill all the fields!");
                    }
                }
                catch(Exception e1) {
                    JOptionPane.showMessageDialog(null, e1);
                }
            }
        });
        btnAddProduct.setForeground(Color.BLACK);
        btnAddProduct.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnAddProduct.setBackground(Color.WHITE);
        btnAddProduct.setBounds(121, 166, 167, 23);
        panel_1.add(btnAddProduct);

        JButton button_1 = new JButton("Clear Contents");
        button_1.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                clearContents();
            }
        });
        button_1.setForeground(Color.BLACK);
        button_1.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_1.setBackground(Color.WHITE);
        button_1.setBounds(298, 166, 167, 23);
        panel_1.add(button_1);
```

```
                JButton button_2 = new JButton("Back");
                button_2.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent e) {
                                dispose();
                        }
                });
                button_2.setForeground(Color.BLACK);
                button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
                button_2.setBackground(Color.WHITE);
                button_2.setBounds(475, 166, 122, 23);
                panel_1.add(button_2);

                JLabel lblQuantity = new JLabel("Rate (in \u20B9):");
                lblQuantity.setForeground(Color.WHITE);
                lblQuantity.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
                lblQuantity.setBounds(363, 72, 93, 23);
                panel_1.add(lblQuantity);

                JLabel lblProductName = new JLabel("Product Name:");
                lblProductName.setForeground(Color.WHITE);
                lblProductName.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
                lblProductName.setBounds(10, 39, 132, 23);
                panel_1.add(lblProductName);

                JLabel lblProductDescription = new JLabel("<HTML>Product 
Description:<HTML>");
                lblProductDescription.setForeground(Color.WHITE);
                lblProductDescription.setFont(new Font("Bahnschrift", Font.PLAIN, 
16));
                lblProductDescription.setBounds(10, 109, 86, 51);
                panel_1.add(lblProductDescription);

                company = new JFormattedTextField();
                company.setBounds(121, 10, 476, 23);
                panel_1.add(company);

                prodname = new JFormattedTextField();
                prodname.setBounds(121, 41, 476, 23);
                panel_1.add(prodname);

                JLabel lblNewLabel_3 = new JLabel("");
                lblNewLabel_3.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
                lblNewLabel_3.setForeground(Color.RED);
                lblNewLabel_3.setBounds(10, 90, 104, 23);
                panel_1.add(lblNewLabel_3);

                size = new JFormattedTextField();
                size.addKeyListener(new KeyAdapter() {
                        @Override
                        public void keyTyped(KeyEvent e) {
                                char vchar = e.getKeyChar();
                                if (vchar == '.' || (vchar == KeyEvent.VK_BACK_SPACE) 
||  (vchar == KeyEvent.VK_DELETE)) {
                                        lblNewLabel_3.setText("");
                                }
                                else {
                                        if ((!(Character.isDigit(vchar)) || (vchar == 
KeyEvent.VK_BACK_SPACE) ||  (vchar == KeyEvent.VK_DELETE))) {
```

```
                        lblNewLabel_3.setText("Invalid Entry!");
                        e.consume();
                    }
                    else {
                        lblNewLabel_3.setText("");
                    }
                }
            }
        });
        size.setBounds(121, 72, 186, 23);
        panel_1.add(size);

        rate = new JFormattedTextField();
        rate.addKeyListener(new KeyAdapter() {
            @Override
            public void keyTyped(KeyEvent e) {
                char vchar = e.getKeyChar();
                if (vchar == '.' || (vchar == KeyEvent.VK_BACK_SPACE)
||  (vchar == KeyEvent.VK_DELETE)) {
                    lblNewLabel_3.setText("");
                }
                else {
                    if ((!(Character.isDigit(vchar)) || (vchar ==
KeyEvent.VK_BACK_SPACE) ||  (vchar == KeyEvent.VK_DELETE))) {
                        lblNewLabel_3.setText("Invalid Entry!");
                        e.consume();
                    }
                    else {
                        lblNewLabel_3.setText("");
                    }
                }
            }
        });
        rate.setBounds(444, 72, 153, 23);
        panel_1.add(rate);

        description = new JTextArea();
        description.setLineWrap(true);
        description.setBounds(121, 109, 476, 51);
        panel_1.add(description);
    }
}
```

4. AddService.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.KeyAdapter;
import java.awt.event.KeyEvent;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;

import javax.swing.JButton;
import javax.swing.JFormattedTextField;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;

public class AddService extends JFrame {

    private JPanel contentPane;
    JFormattedTextField category, subCategory, serviceRate;
    String query, cat, subcat;
    float rate;
    Connection con;
    char vchar;
    PreparedStatement pst;
    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    AddService frame = new AddService();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */

    public void clearContents() {
        category.setText("");
        subCategory.setText("");
        serviceRate.setText("");
    }

    public AddService() {
        setBounds(100, 100, 553, 210);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setLayout(null);
        panel.setBackground(new Color(240, 255, 255));
        panel.setBounds(0, 0, 539, 28);
        contentPane.add(panel);

        JLabel label = new JLabel("Add a new Service");
        label.setForeground(new Color(0, 124, 124));
        label.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
        label.setBounds(0, 0, 692, 26);
        panel.add(label);
        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBackground(Color.DARK_GRAY);
        panel_1.setBounds(0, 30, 539, 143);
        contentPane.add(panel_1);

        JLabel lblCategory = new JLabel("Category:");
        lblCategory.setForeground(Color.WHITE);
        lblCategory.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblCategory.setBounds(10, 8, 86, 23);
        panel_1.add(lblCategory);

        JButton btnAddService = new JButton("Add Service");
        btnAddService.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                query = "INSERT INTO `csiatables`.`servicelist`
(`Category`, `Subcategory`, `Rate`) VALUES (?, ?, ?)";
                cat = category.getText();
                subcat = subCategory.getText();
                rate = Integer.parseInt(serviceRate.getText());
                try {
                    con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                    pst = con.prepareStatement(query);
                    if(category.getText().isEmpty()==false &&
subCategory.getText().isEmpty()==false && serviceRate.getText().isEmpty()==false)
{
                        pst.setString(1, cat);
                        pst.setString(2, subcat);
                        pst.setFloat(3, rate);
                        pst.executeUpdate();
                        JOptionPane.showMessageDialog(null,
"Service added successfully");
                        clearContents();
                    }
                    else {
                        JOptionPane.showMessageDialog(null, "One
or more fields are empty, please fill all the fields!");
                    }
                }
                catch(Exception e1) {

                }
            }
        });
        btnAddService.setForeground(Color.BLACK);
        btnAddService.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnAddService.setBackground(Color.WHITE);
        btnAddService.setBounds(121, 105, 153, 23);
        panel_1.add(btnAddService);

        JButton button_1 = new JButton("Clear Contents");
        button_1.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                clearContents();
            }
        });
        button_1.setForeground(Color.BLACK);
        button_1.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_1.setBackground(Color.WHITE);
        button_1.setBounds(284, 105, 132, 23);
        panel_1.add(button_1);

        JButton button_2 = new JButton("Back");
        button_2.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                dispose();
            }
        });
        button_2.setForeground(Color.BLACK);
        button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_2.setBackground(Color.WHITE);
        button_2.setBounds(426, 105, 74, 23);
        panel_1.add(button_2);

        JLabel label_4 = new JLabel("Rate (in \u20B9):");
        label_4.setForeground(Color.WHITE);
        label_4.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        label_4.setBounds(10, 72, 93, 23);
        panel_1.add(label_4);

        JLabel lblSubcategory = new JLabel("Sub-category:");
        lblSubcategory.setForeground(Color.WHITE);
        lblSubcategory.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblSubcategory.setBounds(10, 39, 132, 23);
        panel_1.add(lblSubcategory);

        category = new JFormattedTextField();
        category.setBounds(121, 10, 379, 23);
        panel_1.add(category);

        subCategory = new JFormattedTextField();
        subCategory.setBounds(121, 41, 379, 23);
        panel_1.add(subCategory);

        JLabel lblNewLabel_3 = new JLabel("");
        lblNewLabel_3.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblNewLabel_3.setForeground(Color.RED);
        lblNewLabel_3.setBounds(217, 72, 104, 23);
        panel_1.add(lblNewLabel_3);

        serviceRate = new JFormattedTextField();
        serviceRate.addKeyListener(new KeyAdapter() {
            @Override
            public void keyTyped(KeyEvent e) {
                vchar = e.getKeyChar();
                if (vchar == '.' || (vchar == KeyEvent.VK_BACK_SPACE)
|| (vchar == KeyEvent.VK_DELETE)) {
                    lblNewLabel_3.setText("");
                }
                else {
                    if ((!(Character.isDigit(vchar)) || (vchar ==
KeyEvent.VK_BACK_SPACE) || (vchar == KeyEvent.VK_DELETE))) {
                        lblNewLabel_3.setText("Invalid Entry!");
                        e.consume();
                    }
                    else {
                        lblNewLabel_3.setText("");
                    }
                }

            }
        });
        serviceRate.setBounds(121, 72, 86, 23);
        panel_1.add(serviceRate);
    }
}
```

5. Branches.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;
import javax.swing.table.DefaultTableModel;
import javax.swing.table.TableColumnModel;
import javax.swing.table.TableModel;

public class Branches extends JFrame {

    private JPanel contentPane;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Branches frame = new Branches();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */

    public Branches() {
        super("Beautymanntra || Branches");
        setResizable(false);
        setBounds(100, 100, 1138, 203);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
```

```
            contentPane.setBackground(new Color(240, 255, 255));
            setContentPane(contentPane);
            contentPane.setLayout(null);

            JPanel panel_1 = new JPanel();
            panel_1.setLayout(null);
            panel_1.setBackground(new Color(240, 255, 255));
            panel_1.setBounds(0, 0, 1134, 79);
            contentPane.add(panel_1);

            JLabel lblBeautymanntraBranches = new JLabel("Beautymanntra
Branches");

      lblBeautymanntraBranches.setHorizontalAlignment(SwingConstants.CENTER);
            lblBeautymanntraBranches.setForeground(new Color(0, 124, 124));
            lblBeautymanntraBranches.setFont(new Font("Bahnschrift", Font.PLAIN,
60));
            lblBeautymanntraBranches.setBounds(0, 0, 1133, 80);
            panel_1.add(lblBeautymanntraBranches);

            JPanel panel = new JPanel();
            panel.setBackground(Color.DARK_GRAY);
            panel.setBounds(0, 78, 1134, 97);
            contentPane.add(panel);
            panel.setLayout(null);

            JButton btnAddANew = new JButton("Add a new Branch");
            btnAddANew.addActionListener(new ActionListener() {
                  public void actionPerformed(ActionEvent e) {
                        AddBranch a = new AddBranch();
                        a.setVisible(true);
                  }
            });
            btnAddANew.setForeground(Color.BLACK);
            btnAddANew.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
            btnAddANew.setBackground(Color.WHITE);
            btnAddANew.setBounds(10, 10, 211, 35);
            panel.add(btnAddANew);

            JButton btnViewListOf = new JButton("View List of Branches");
            btnViewListOf.addActionListener(new ActionListener() {
                  public void actionPerformed(ActionEvent e) {
                        DisplayTableBranch d = new DisplayTableBranch();
                        d.setVisible(true);
                  }
            });
            btnViewListOf.setForeground(Color.BLACK);
            btnViewListOf.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
            btnViewListOf.setBackground(Color.WHITE);
            btnViewListOf.setBounds(231, 10, 211, 35);
            panel.add(btnViewListOf);

            JButton button = new JButton("Back");
            button.addActionListener(new ActionListener() {
                  public void actionPerformed(ActionEvent e) {
                        dispose();
                        MainMenu mm = new MainMenu();
                        mm.setVisible(true);
                  }
```

```
        });
        button.setForeground(Color.BLACK);
        button.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button.setBackground(Color.WHITE);
        button.setBounds(894, 10, 211, 35);
        panel.add(button);

        JButton btnEditAnExisting = new JButton("Edit an existing Branch");
        btnEditAnExisting.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                EditBranch eb = new EditBranch();
                eb.setVisible(true);
            }
        });
        btnEditAnExisting.setForeground(Color.BLACK);
        btnEditAnExisting.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnEditAnExisting.setBackground(Color.WHITE);
        btnEditAnExisting.setBounds(452, 10, 211, 35);
        panel.add(btnEditAnExisting);

        JButton btnDeleteABranch = new JButton("Delete a Branch");
        btnDeleteABranch.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                DeleteBranch db = new DeleteBranch();
                db.setVisible(true);
            }
        });
        btnDeleteABranch.setForeground(Color.BLACK);
        btnDeleteABranch.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnDeleteABranch.setBackground(Color.WHITE);
        btnDeleteABranch.setBounds(673, 10, 211, 35);
        panel.add(btnDeleteABranch);
    }
}
```

6. Clients.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;

public class Clients extends JFrame {

    private JPanel contentPane;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
                public void run() {
                    try {
                        Clients frame = new Clients();
                        frame.setVisible(true);
                    } catch (Exception e) {
                        e.printStackTrace();
                    }
                }
            });
    }

    /**
     * Create the frame.
     */
    public Clients() {
        super("Beautymanntra || Clients");
        setResizable(false);
        setBounds(100, 100, 903, 203);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setLayout(null);
        panel.setBackground(new Color(240, 255, 255));
        panel.setBounds(0, 0, 1134, 79);
        contentPane.add(panel);

        JLabel lblClients = new JLabel("Clients");
        lblClients.setHorizontalAlignment(SwingConstants.CENTER);
        lblClients.setForeground(new Color(0, 124, 124));
        lblClients.setFont(new Font("Bahnschrift", Font.PLAIN, 60));
        lblClients.setBounds(0, 0, 900, 80);
        panel.add(lblClients);

        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBackground(Color.DARK_GRAY);
        panel_1.setBounds(0, 78, 1134, 97);
        contentPane.add(panel_1);

        JButton btnAddANew = new JButton("Add a new Client");
        btnAddANew.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                AddClient ab = new AddClient();
                ab.setVisible(true);
            }
        });
        btnAddANew.setForeground(Color.BLACK);
        btnAddANew.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnAddANew.setBackground(Color.WHITE);
        btnAddANew.setBounds(10, 10, 211, 35);
        panel_1.add(btnAddANew);

        JButton btnViewListOf = new JButton("View List of Clients");
        btnViewListOf.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                DisplayTableClient d = new DisplayTableClient();
                d.setVisible(true);
            }
        });
        btnViewListOf.setForeground(Color.BLACK);
        btnViewListOf.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnViewListOf.setBackground(Color.WHITE);
        btnViewListOf.setBounds(231, 10, 211, 35);
        panel_1.add(btnViewListOf);

        JButton button_2 = new JButton("Back");
        button_2.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                MainMenu mm = new MainMenu();
                mm.setVisible(true);
                dispose();
            }
        });
        button_2.setForeground(Color.BLACK);
        button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_2.setBackground(Color.WHITE);
        button_2.setBounds(673, 10, 211, 35);
        panel_1.add(button_2);

        JButton btnEditAnExisting = new JButton("Edit an existing Client");
        btnEditAnExisting.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                EditClient e1 = new EditClient();
                e1.setVisible(true);
            }
        });
        btnEditAnExisting.setForeground(Color.BLACK);
        btnEditAnExisting.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnEditAnExisting.setBackground(Color.WHITE);
        btnEditAnExisting.setBounds(452, 10, 211, 35);
        panel_1.add(btnEditAnExisting);
    }
}
```

7. DeleteBranch.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

import javax.swing.JButton;
import javax.swing.JFormattedTextField;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
```

```
public class DeleteBranch extends JFrame {

    private JPanel contentPane;
    JFormattedTextField branchID;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    DeleteBranch frame = new DeleteBranch();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */

    PreparedStatement pst;
    ResultSet rs;
    String query;
    Connection con;
    public DeleteBranch() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 551, 155);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setLayout(null);
        panel.setBorder(new EmptyBorder(5, 5, 5, 5));
        panel.setBounds(0, 0, 863, 189);
        contentPane.add(panel);

        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBackground(new Color(240, 255, 255));
        panel_1.setBounds(0, 0, 1108, 36);
        panel.add(panel_1);

        JLabel lblDeleteABranch = new JLabel("Delete a Branch");
        lblDeleteABranch.setForeground(new Color(0, 124, 124));
        lblDeleteABranch.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
        lblDeleteABranch.setBounds(10, 0, 1098, 36);
        panel_1.add(lblDeleteABranch);

        JPanel panel_2 = new JPanel();
        panel_2.setLayout(null);
        panel_2.setBackground(Color.DARK_GRAY);
        panel_2.setBounds(0, 31, 1108, 181);
        panel.add(panel_2);

        JLabel label_1 = new JLabel("Branch ID:");
        label_1.setForeground(Color.WHITE);
        label_1.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        label_1.setBounds(10, 21, 132, 23);
        panel_2.add(label_1);

        branchID = new JFormattedTextField();
        branchID.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        branchID.setColumns(10);
        branchID.setBounds(152, 23, 372, 23);
        panel_2.add(branchID);

        JButton btnDeleteBranch = new JButton("Delete Branch");
        btnDeleteBranch.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                try {
                    con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                    query = "DELETE FROM `csiatables`.`branchlist`
WHERE (`BranchID` = ? )";
                    pst = con.prepareStatement(query);
                    pst.setString(1, branchID.getText());
                    pst.executeUpdate();
                    JOptionPane.showMessageDialog(null, "Branch
deleted successfully!");
                } catch (SQLException e1) {
                    // TODO Auto-generated catch block
                    e1.printStackTrace();
                }
            }
        });
        btnDeleteBranch.setForeground(Color.BLACK);
        btnDeleteBranch.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnDeleteBranch.setBackground(Color.WHITE);
        btnDeleteBranch.setBounds(152, 56, 132, 23);
        panel_2.add(btnDeleteBranch);

        JButton btnBack = new JButton("Back");
        btnBack.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                dispose();
            }
        });
        btnBack.setForeground(Color.BLACK);
        btnBack.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnBack.setBackground(Color.WHITE);
        btnBack.setBounds(427, 56, 97, 23);
        panel_2.add(btnBack);

        JButton btnClearField = new JButton("Clear Field");
        btnClearField.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                branchID.setText("");
            }
        });
        btnClearField.setForeground(Color.BLACK);
```

```
		btnClearField.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
		btnClearField.setBackground(Color.WHITE);
		btnClearField.setBounds(294, 56, 123, 23);
		panel_2.add(btnClearField);
	}
}
```

8. DeleteProduct.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

import javax.swing.JButton;
import javax.swing.JFormattedTextField;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;

public class DeleteProduct extends JFrame {

	private JPanel contentPane;
	JFormattedTextField serialNo;
	JLabel company, pname, description, rate, size;
	Connection con;
	String query, company1, pname1, description1, size1, rate1;
	ResultSet rs;
	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {
				try {
					DeleteProduct frame = new DeleteProduct();
					frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}

	/**
	 * Create the frame.
	 */



	public void clearContents() {
		serialNo.setText("");
```

```
            company.setText("");
            description.setText("");
            size.setText("");
            rate.setText("");
            pname.setText("");
    }

    PreparedStatement pst;
    public DeleteProduct() {
            setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            setBounds(100, 100, 397, 315);
            contentPane = new JPanel();
            contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
            setContentPane(contentPane);
            contentPane.setLayout(null);

            JPanel panel = new JPanel();
            panel.setBounds(0, 0, 383, 278);
            contentPane.add(panel);
            panel.setLayout(null);

            JPanel panel_1 = new JPanel();
            panel_1.setLayout(null);
            panel_1.setBackground(new Color(240, 255, 255));
            panel_1.setBounds(0, 0, 383, 36);
            panel.add(panel_1);

            JLabel lblDeleteAProduct = new JLabel("Delete a Product\r\n");
            lblDeleteAProduct.setForeground(new Color(0, 124, 124));
            lblDeleteAProduct.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
            lblDeleteAProduct.setBounds(0, 0, 383, 26);
            panel_1.add(lblDeleteAProduct);

            JPanel panel_2 = new JPanel();
            panel_2.setLayout(null);
            panel_2.setBackground(Color.DARK_GRAY);
            panel_2.setBounds(0, 35, 383, 244);
            panel.add(panel_2);

            JLabel label_1 = new JLabel("Serial Number: ");
            label_1.setForeground(Color.WHITE);
            label_1.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
            label_1.setBounds(10, 10, 83, 45);
            panel_2.add(label_1);

            JLabel label_2 = new JLabel("Product Name:");
            label_2.setForeground(Color.WHITE);
            label_2.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
            label_2.setBounds(10, 78, 78, 45);
            panel_2.add(label_2);

            JButton button = new JButton("Back");
            button.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                            dispose();
                    }
            });
            button.setForeground(Color.BLACK);
            button.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
```

```java
button.setBackground(Color.WHITE);
button.setBounds(236, 211, 102, 23);
panel_2.add(button);

JLabel label_3 = new JLabel("<HTML>Product Description:<HTML>");
label_3.setForeground(Color.WHITE);
label_3.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
label_3.setBounds(10, 116, 75, 51);
panel_2.add(label_3);

JLabel label_4 = new JLabel("Size:");
label_4.setForeground(Color.WHITE);
label_4.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
label_4.setBounds(10, 177, 25, 15);
panel_2.add(label_4);

JLabel label_5 = new JLabel("Rate: ");
label_5.setForeground(Color.WHITE);
label_5.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
label_5.setBounds(168, 177, 30, 15);
panel_2.add(label_5);

JLabel label_6 = new JLabel("Company:");
label_6.setForeground(Color.WHITE);
label_6.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
label_6.setBounds(10, 40, 51, 45);
panel_2.add(label_6);

company = new JLabel("");
company.setForeground(Color.GREEN);
company.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
company.setBounds(74, 40, 299, 45);
panel_2.add(company);

pname = new JLabel("");
pname.setForeground(Color.GREEN);
pname.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
pname.setBounds(98, 78, 177, 45);
panel_2.add(pname);

description = new JLabel("<HTML><HTML>");
description.setForeground(Color.GREEN);
description.setFont(new Font("Bahnschrift", Font.PLAIN, 10));
description.setBounds(85, 116, 288, 51);
panel_2.add(description);

size = new JLabel("");
size.setForeground(Color.GREEN);
size.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
size.setBounds(41, 169, 117, 31);
panel_2.add(size);

rate = new JLabel("");
rate.setForeground(Color.GREEN);
rate.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
rate.setBounds(220, 169, 134, 31);
panel_2.add(rate);

serialNo = new JFormattedTextField();
```

```
             serialNo.setBounds(117, 24, 146, 19);
             panel_2.add(serialNo);

             JButton btnDeleteProduct = new JButton("Delete Product");
             btnDeleteProduct.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                          query = "DELETE FROM `csiatables`.`productlist` WHERE
(`SerialNo` = ? )";
                          try {
                                 con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                                 pst = con.prepareStatement(query);
                                 pst.setInt(1,
Integer.parseInt(serialNo.getText()));
                                 pst.executeUpdate();
                                 JOptionPane.showMessageDialog(null, "Product
deleted successfully!");
                                 clearContents();
                          }
                          catch(Exception e1) {
                                 JOptionPane.showMessageDialog(null, e1);
                          }
                    }
             });
             btnDeleteProduct.setForeground(Color.BLACK);
             btnDeleteProduct.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
             btnDeleteProduct.setBackground(Color.WHITE);
             btnDeleteProduct.setBounds(90, 211, 136, 23);
             panel_2.add(btnDeleteProduct);

             JButton btnSearch = new JButton("Search");
             btnSearch.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                          Function f = new Function();
                          rs = null;
                          company1 = "Company";
                          pname1 = "ProductName";
                          description1 = "ProductDescription";
                          size1 = "Size";
                          rate1 = "Rate";
                          if(serialNo.getText().isEmpty() == true) {
                                 JOptionPane.showMessageDialog(null, "Field Blank!
Please enter a Client ID!");
                          }
                          else {
                                 rs=f.find(Integer.parseInt(serialNo.getText()));
                                 try {
                                        if (rs.next()) {

      company.setText(rs.getString(company1));
                                              pname.setText(rs.getString(pname1));

      description.setText(rs.getString(description1));

      size.setText(Integer.toString(rs.getInt(size1)));

      rate.setText(Integer.toString(rs.getInt(rate1)));
                                              serialNo.setEditable(false);
```

```
                        }
                        else {
                            JOptionPane.showMessageDialog(null,
"No product registered with that serial number");
                            serialNo.setText("");
                        }
                    }
                    catch(Exception e1) {

                    }
                }
            }
        });
        btnSearch.setForeground(Color.BLACK);
        btnSearch.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnSearch.setBackground(Color.WHITE);
        btnSearch.setBounds(271, 23, 83, 23);
        panel_2.add(btnSearch);
    }
}
```

9. DeleteService.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

import javax.swing.JButton;
import javax.swing.JFormattedTextField;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;

public class DeleteService extends JFrame {

    private JPanel contentPane;
    JFormattedTextField srlno;
    JLabel cat, subcat, rate;
    String query, cat1, subcat1, rate1;
    ResultSet rs;
    Connection con;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    DeleteService frame = new DeleteService();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */



    public void clearContents() {
        srlno.setEditable(true);
        srlno.setText("");
        cat.setText("");
        subcat.setText("");
        rate.setText("");
    }

    PreparedStatement pst;

    public DeleteService() {
        setBounds(100, 100, 553, 237);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setLayout(null);
        panel.setBackground(new Color(240, 255, 255));
        panel.setBounds(0, 0, 612, 28);
        contentPane.add(panel);

        JLabel lblDeleteAnExisting = new JLabel("Delete an existing
Service");
        lblDeleteAnExisting.setForeground(new Color(0, 124, 124));
        lblDeleteAnExisting.setFont(new Font("Bahnschrift", Font.PLAIN,
24));
        lblDeleteAnExisting.setBounds(0, 0, 537, 28);
        panel.add(lblDeleteAnExisting);

        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBackground(Color.DARK_GRAY);
        panel_1.setBounds(0, 30, 539, 186);
        contentPane.add(panel_1);

        JLabel label_1 = new JLabel("Category:");
        label_1.setForeground(Color.WHITE);
        label_1.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        label_1.setBounds(10, 38, 86, 23);
        panel_1.add(label_1);

        JButton button = new JButton("Delete Service");
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
```

```
                          query = "DELETE FROM `csiatables`.`servicelist` WHERE
(`SerialNo` = ? )";
                          try {
                                 con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                                 pst = con.prepareStatement(query);
                                 pst.setInt(1, Integer.parseInt(srlno.getText()));
                                 pst.executeUpdate();
                                 JOptionPane.showMessageDialog(null, "Service
removed successfully!");
                                 clearContents();
                          }
                          catch(Exception e1) {
                                 JOptionPane.showMessageDialog(null, e1);
                          }
                    }
             });
             button.setForeground(Color.BLACK);
             button.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
             button.setBackground(Color.WHITE);
             button.setBounds(121, 135, 153, 23);
             panel_1.add(button);

             JButton button_1 = new JButton("Clear Contents");
             button_1.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                          clearContents();
                    }
             });
             button_1.setForeground(Color.BLACK);
             button_1.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
             button_1.setBackground(Color.WHITE);
             button_1.setBounds(284, 135, 132, 23);
             panel_1.add(button_1);

             JButton button_2 = new JButton("Back");
             button_2.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                          dispose();
                    }
             });
             button_2.setForeground(Color.BLACK);
             button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
             button_2.setBackground(Color.WHITE);
             button_2.setBounds(426, 135, 74, 23);
             panel_1.add(button_2);

             JLabel label_2 = new JLabel("Rate (in \u20B9):");
             label_2.setForeground(Color.WHITE);
             label_2.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
             label_2.setBounds(10, 102, 93, 23);
             panel_1.add(label_2);

             JLabel label_3 = new JLabel("Sub-category:");
             label_3.setForeground(Color.WHITE);
             label_3.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
             label_3.setBounds(10, 69, 132, 23);
             panel_1.add(label_3);
```

```
        JLabel label_5 = new JLabel("Serial Number:");
        label_5.setForeground(Color.WHITE);
        label_5.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        label_5.setBounds(10, 10, 110, 23);
        panel_1.add(label_5);

        srlno = new JFormattedTextField();
        srlno.setBounds(121, 7, 153, 23);
        panel_1.add(srlno);

        JButton button_3 = new JButton("Search");
        button_3.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                Function f = new Function();
                rs = null;
                cat1 = "Category";
                subcat1 = "Subcategory";
                rate1 = "Rate";
                if(srlno.getText().isEmpty() == true) {
                    JOptionPane.showMessageDialog(null, "Field Blank!
Please enter the Service Serial No.!");
                }
                else {
                    rs=f.find1(Integer.parseInt(srlno.getText()));
                    try {
                        if (rs.next()) {
                            cat.setText(rs.getString(cat1));

    subcat.setText(rs.getString(subcat1));

    rate.setText(Integer.toString(rs.getInt(rate1)));
                            srlno.setEditable(false);
                        }
                        else {
                            JOptionPane.showMessageDialog(null,
"No service registered with that serial number");
                        }
                    }
                    catch(Exception e1) {

                    }
                }
            }
        });
        button_3.setForeground(Color.BLACK);
        button_3.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_3.setBackground(Color.WHITE);
        button_3.setBounds(282, 7, 153, 23);
        panel_1.add(button_3);

        cat = new JLabel("");
        cat.setForeground(Color.GREEN);
        cat.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        cat.setBounds(121, 38, 379, 23);
        panel_1.add(cat);

        subcat = new JLabel("");
        subcat.setForeground(Color.GREEN);
        subcat.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        subcat.setBounds(121, 69, 379, 23);
        panel_1.add(subcat);

        rate = new JLabel("");
        rate.setForeground(Color.GREEN);
        rate.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        rate.setBounds(121, 102, 379, 23);
        panel_1.add(rate);
    }

}
```

10. DisplayTableBranch.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;

import net.proteanit.sql.DbUtils;

public class DisplayTableBranch extends JFrame {

    private JPanel contentPane;
    static JTable table;
    Connection con, conn;
    String query;
    int row;
    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    DisplayTableBranch frame = new
DisplayTableBranch();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */

    static void updateTable() {
        String query;
        Connection conn;
        try {
            conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
            query = "select * from branchlist";
            pst = conn.prepareStatement(query);
            rs = pst.executeQuery();
            table.setModel(DbUtils.resultSetToTableModel(rs));
        } catch (SQLException e1) {
            // TODO Auto-generated catch block
            e1.printStackTrace();
        }
    }

    static PreparedStatement pst;
    static ResultSet rs;
    public DisplayTableBranch() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 300, 1339, 403);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setBackground(Color.CYAN.darker().darker().darker());
        panel.setBounds(0, 0, 1325, 366);
        contentPane.add(panel);
        panel.setLayout(null);

        JButton btnDeleteSelectedBranch1 = new JButton("Delete Selected
Branch");
        btnDeleteSelectedBranch1.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                try {
                    row = table.getSelectedRow();
                    con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                    query = "DELETE FROM branchlist where BranchID =
'" + table.getModel().getValueAt(row, 0) + "'";
                    pst = con.prepareStatement(query);
                    pst.execute();
                    JOptionPane.showMessageDialog(null, "Branch
deleted successfully!");
                    updateTable();
                    btnDeleteSelectedBranch1.setVisible(false);
                } catch (SQLException e1) {
                    JOptionPane.showMessageDialog(null, "Error");
                }
            }
        });
        btnDeleteSelectedBranch1.setForeground(Color.BLACK);
        btnDeleteSelectedBranch1.setFont(new Font("Bahnschrift", Font.PLAIN,
14));
        btnDeleteSelectedBranch1.setBackground(Color.WHITE);
        btnDeleteSelectedBranch1.setBounds(113, 5, 227, 25);
        btnDeleteSelectedBranch1.setVisible(false);
        panel.add(btnDeleteSelectedBranch1);

        JLabel l = new JLabel("Beautymanntra Branches");
        l.setBounds(104, 5, 1117, 25);
        l.setHorizontalAlignment(SwingConstants.CENTER);
        l.setFont(new Font("Bahnschrift", Font.PLAIN, 26));
        l.setForeground(Color.WHITE);
        panel.add(l);

        JScrollPane scrollPane = new JScrollPane();
        scrollPane.setBounds(10, 33, 1305, 323);
        panel.add(scrollPane);

        table = new JTable();
        table.addMouseListener(new MouseAdapter() {
            @Override
            public void mouseClicked(MouseEvent e) {
                btnDeleteSelectedBranch1.setVisible(true);
            }
        });
        table.setDefaultEditor(Object.class, null);
        table.setRowHeight(50);
        scrollPane.setViewportView(table);

        JButton btnBack = new JButton("Back");
        btnBack.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                dispose();
            }
        });
        btnBack.setForeground(Color.BLACK);
        btnBack.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnBack.setBackground(Color.WHITE);
        btnBack.setBounds(1222, 5, 93, 25);
        panel.add(btnBack);

        JButton btnView = new JButton("View");
        btnView.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                try {
                    conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                    query = "select * from branchlist order by
BranchID desc";

                    pst = conn.prepareStatement(query);
                    rs = pst.executeQuery();
```

```
    table.setModel(DbUtils.resultSetToTableModel(rs));
                } catch (SQLException e1) {
                    JOptionPane.showMessageDialog(null, "Error");
                }
            }
        });
        btnView.setForeground(Color.BLACK);
        btnView.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnView.setBackground(Color.WHITE);
        btnView.setBounds(10, 5, 93, 25);
        panel.add(btnView);
    }
}
```

11. DisplayTableClient.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;
import javax.swing.table.TableColumnModel;

import net.proteanit.sql.DbUtils;

public class DisplayTableClient extends JFrame {

    private JPanel contentPane;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    DisplayTableClient frame = new
DisplayTableClient();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */

    PreparedStatement pst;
    ResultSet rs;
    Connection conn;
    String query;
    private JTable table;
    TableColumnModel tcm;
    int row;
    public DisplayTableClient() {
        setBounds(100, 300, 1339, 403);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setBounds(0, 0, 1325, 366);
        contentPane.add(panel);
        panel.setBackground(Color.CYAN.darker().darker().darker());
        panel.setLayout(null);

        JButton btnViewClientInformation = new JButton("View Client 
Information");
        btnViewClientInformation.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                ViewClientInfo v = new ViewClientInfo();
                v.setVisible(true);
                row = table.getSelectedRow();
                v.clientLabel.setText(table.getModel().getValueAt(row, 
0) + "");

                v.nameLabel.setText(table.getModel().getValueAt(row, 1) 
+ "");

                v.numberLabel.setText(table.getModel().getValueAt(row, 
2) + "");

                v.emailLabel.setText(table.getModel().getValueAt(row, 
3) + "");

                v.genderLabel.setText(table.getModel().getValueAt(row, 
4) + "");

                v.bdayLabel.setText(table.getModel().getValueAt(row, 5) 
+ "");
            }
        });
        btnViewClientInformation.setForeground(Color.BLACK);
        btnViewClientInformation.setVisible(false);
        btnViewClientInformation.setFont(new Font("Bahnschrift", Font.PLAIN, 
14));

        btnViewClientInformation.setBackground(Color.WHITE);
        btnViewClientInformation.setBounds(148, 5, 208, 25);
        panel.add(btnViewClientInformation);

        JButton btnViewList = new JButton("View List");
        btnViewList.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                try {
                    conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                    query = "select * from clientlist order by Name
ASC";
                    pst = conn.prepareStatement(query);
                    rs = pst.executeQuery();

    table.setModel(DbUtils.resultSetToTableModel(rs));
                    tcm = table.getColumnModel();
                    tcm.removeColumn(tcm.getColumn(5));
                    tcm.removeColumn(tcm.getColumn(4));
                    tcm.removeColumn(tcm.getColumn(3));
                } catch (SQLException e1) {
                    JOptionPane.showMessageDialog(null, e1);
                }
            }
        });
        btnViewList.setBounds(10, 5, 128, 25);
        btnViewList.setForeground(Color.BLACK);
        btnViewList.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnViewList.setBackground(Color.WHITE);
        panel.add(btnViewList);

        JButton button_1 = new JButton("Back");
        button_1.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                dispose();
            }
        });
        button_1.setForeground(Color.BLACK);
        button_1.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_1.setBackground(Color.WHITE);
        button_1.setBounds(1222, 5, 91, 25);
        panel.add(button_1);

        JLabel lblListOfClients = new JLabel("List of Clients");
        lblListOfClients.setBounds(373, 0, 609, 30);
        lblListOfClients.setHorizontalAlignment(SwingConstants.CENTER);
        lblListOfClients.setForeground(Color.WHITE);
        lblListOfClients.setFont(new Font("Bahnschrift", Font.PLAIN, 26));
        panel.add(lblListOfClients);

        JScrollPane scrollPane = new JScrollPane();
        scrollPane.setBounds(10, 34, 1303, 322);
        panel.add(scrollPane);

        table = new JTable();
        table.addMouseListener(new MouseAdapter() {
            @Override
            public void mouseClicked(MouseEvent e) {
                btnViewClientInformation.setVisible(true);
            }
        });
        table.setDefaultEditor(Object.class, null);
        scrollPane.setViewportView(table);
```

```
		table.setRowHeight(30);
	}
}
```

12. DisplayTableInvoice.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.io.File;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;

import net.proteanit.sql.DbUtils;

public class DisplayTableInvoice extends JFrame {

	private JPanel contentPane;
	private JTable table;
	Connection con;
	String query;
	PreparedStatement pst;
	ResultSet rs;
	File file;
	String path;
	Process pro;
	int row;
	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {
				try {
					DisplayTableInvoice frame = new 
DisplayTableInvoice();
					frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}
```

```
    /**
     * Create the frame.
     */
    public DisplayTableInvoice() {
        setBounds(100, 300, 1339, 403);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setBackground(Color.CYAN.darker().darker().darker());
        panel.setBounds(0, 0, 1325, 366);
        contentPane.add(panel);
        panel.setLayout(null);

        JButton button = new JButton("View List");
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                try {
                    con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                    query = "select SerialNo, Path, ClientName, Date
from invoicelist order by Date DESC";
                    pst = con.prepareStatement(query);
                    rs = pst.executeQuery();

    table.setModel(DbUtils.resultSetToTableModel(rs));
                }
                catch(Exception e1) {
                    JOptionPane.showMessageDialog(null, e1);
                }
            }
        });
        button.setForeground(Color.BLACK);
        button.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button.setBackground(Color.WHITE);
        button.setBounds(10, 5, 128, 25);
        panel.add(button);

        JButton btnOpenInvoice = new JButton("Open Invoice");
        btnOpenInvoice.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                row = table.getSelectedRow();
                try {
                    path =(String) table.getModel().getValueAt(row,
1);
                    file = new File(path);
                    if (file.exists()) {
                        pro = Runtime.getRuntime().exec("rundll32
url.dll, FileProtocolHandler " + path);
                        pro.waitFor();

                    }
                    else {
                        JOptionPane.showMessageDialog(null,
"Invoice does not exist");
```

```
                    }
                }
                catch(Exception e1) {
                    
                }
            }
        });
        btnOpenInvoice.setForeground(Color.BLACK);
        btnOpenInvoice.setVisible(false);
        btnOpenInvoice.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnOpenInvoice.setBackground(Color.WHITE);
        btnOpenInvoice.setBounds(148, 5, 128, 25);
        panel.add(btnOpenInvoice);
        
        JLabel lblListOfInvoices = new JLabel("List of Invoices");
        lblListOfInvoices.setHorizontalAlignment(SwingConstants.CENTER);
        lblListOfInvoices.setForeground(Color.WHITE);
        lblListOfInvoices.setFont(new Font("Bahnschrift", Font.PLAIN, 26));
        lblListOfInvoices.setBounds(373, 0, 609, 30);
        panel.add(lblListOfInvoices);
        
        JButton button_1 = new JButton("Back");
        button_1.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                dispose();
                Invoices ii = new Invoices();
                ii.setVisible(true);
            }
        });
        button_1.setForeground(Color.BLACK);
        button_1.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_1.setBackground(Color.WHITE);
        button_1.setBounds(1222, 5, 91, 25);
        panel.add(button_1);
        
        JScrollPane scrollPane = new JScrollPane();
        scrollPane.setBounds(10, 34, 1303, 322);
        panel.add(scrollPane);
        
        table = new JTable();
        table.setDefaultEditor(Object.class, null);
        table.addMouseListener(new MouseAdapter() {
            @Override
            public void mouseClicked(MouseEvent e) {
                btnOpenInvoice.setVisible(true);
            }
        });
        scrollPane.setViewportView(table);
        table.setRowHeight(30);
    }
}
```

13. DisplayTableProduct.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;
import javax.swing.table.TableColumnModel;

import net.proteanit.sql.DbUtils;

public class DisplayTableProduct extends JFrame {

    private JPanel contentPane;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    DisplayTableProduct frame = new
DisplayTableProduct();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    Connection conn;
    String query;
    int row;
    PreparedStatement pst;
    ResultSet rs;
    TableColumnModel tcm;
    private JTable table;
    public DisplayTableProduct() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 300, 1339, 403);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);
        JPanel panel = new JPanel();
        panel.setBorder(new EmptyBorder(5, 5, 5, 5));
        panel.setBounds(0, 0, 1325, 366);
        contentPane.add(panel);
        panel.setBackground(Color.CYAN.darker().darker().darker());
        panel.setLayout(null);

        JLabel lblListOfClients = new JLabel("List of Products");
        lblListOfClients.setBounds(365, 0, 599, 30);
        lblListOfClients.setHorizontalAlignment(SwingConstants.CENTER);
        lblListOfClients.setForeground(Color.WHITE);
        lblListOfClients.setFont(new Font("Bahnschrift", Font.PLAIN, 26));
        panel.add(lblListOfClients);


        JButton btnViewProductInformation = new JButton("View Product
Information\r\n");
        btnViewProductInformation.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                ViewProductInfo v = new ViewProductInfo();
                v.setVisible(true);
                row = table.getSelectedRow();
                v.serialno.setText(table.getModel().getValueAt(row, 0)
+ "");
                v.company.setText(table.getModel().getValueAt(row, 1) +
"");
                v.pname.setText("<HTML>" +
table.getModel().getValueAt(row, 2) + "<HTML>");
                v.description.setText("<HTML>" +
table.getModel().getValueAt(row, 3) + "<HTML>");
                v.size.setText(table.getModel().getValueAt(row, 4) +
"");
                v.rate.setText(table.getModel().getValueAt(row, 5) +
"");
            }
        });
        btnViewProductInformation.setVisible(false);
        btnViewProductInformation.setForeground(Color.BLACK);
        btnViewProductInformation.setFont(new Font("Bahnschrift",
Font.PLAIN, 14));
        btnViewProductInformation.setBackground(Color.WHITE);
        btnViewProductInformation.setBounds(151, 5, 204, 25);
        panel.add(btnViewProductInformation);

        JScrollPane scrollPane = new JScrollPane();
        scrollPane.setBounds(10, 34, 1303, 322);
        panel.add(scrollPane);

        table = new JTable();
        table.addMouseListener(new MouseAdapter() {
            @Override
            public void mouseClicked(MouseEvent e) {
                btnViewProductInformation.setVisible(true);
            }
        });
        table.setDefaultEditor(Object.class, null);
        scrollPane.setViewportView(table);
        table.setRowHeight(30);
        JButton button = new JButton("View List");
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                try {
                    conn =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                    query = "select * from productlist";
                    pst = conn.prepareStatement(query);
                    rs = pst.executeQuery();

    table.setModel(DbUtils.resultSetToTableModel(rs));
                    tcm = table.getColumnModel();
                    tcm.removeColumn(tcm.getColumn(5));
                    tcm.removeColumn(tcm.getColumn(4));
                }
                catch(Exception e1) {
                    JOptionPane.showMessageDialog(null, e1);
                }
            }
        });
        button.setForeground(Color.BLACK);
        button.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button.setBackground(Color.WHITE);
        button.setBounds(10, 5, 128, 25);
        panel.add(button);

        JButton btnBack = new JButton("Back");
        btnBack.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                dispose();
            }
        });
        btnBack.setForeground(Color.BLACK);
        btnBack.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnBack.setBackground(Color.WHITE);
        btnBack.setBounds(1185, 5, 128, 25);
        panel.add(btnBack);


    }
}
```

14. DisplayTableService.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;
import javax.swing.table.TableColumnModel;

import net.proteanit.sql.DbUtils;

public class DisplayTableService extends JFrame {

	private JPanel contentPane;

	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {
				try {
					DisplayTableService frame = new
DisplayTableService();
					frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}

	/**
	 * Create the frame.
	 */
	PreparedStatement pst;
	ResultSet rs;
	TableColumnModel tcm;
	private JTable table;
	Connection conn;
	String query;

	public DisplayTableService() {
		setBounds(100, 300, 1339, 403);
		contentPane = new JPanel();
		contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
		setContentPane(contentPane);
		contentPane.setLayout(null);

		JPanel panel = new JPanel();
		panel.setBackground(Color.CYAN.darker().darker().darker());
		panel.setBounds(0, 0, 1325, 366);
		contentPane.add(panel);
		panel.setLayout(null);

		JScrollPane scrollPane = new JScrollPane();
		scrollPane.setBounds(10, 34, 1303, 322);
		panel.add(scrollPane);

		table = new JTable();
		table.setDefaultEditor(Object.class, null);
```

```
			scrollPane.setViewportView(table);
			table.setRowHeight(30);

			JButton button = new JButton("View List");
			button.addActionListener(new ActionListener() {
				public void actionPerformed(ActionEvent e) {
					try {
						conn = 
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root", 
"mySQLpassword");
						query = "select * from servicelist";
						pst = conn.prepareStatement(query);
						rs = pst.executeQuery();
						
		table.setModel(DbUtils.resultSetToTableModel(rs));
					} catch (SQLException e1) {
						// TODO Auto-generated catch block
						e1.printStackTrace();
					}
				}
			});
			button.setForeground(Color.BLACK);
			button.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
			button.setBackground(Color.WHITE);
			button.setBounds(10, 5, 128, 25);
			panel.add(button);

			JLabel lblListOfServices = new JLabel("List of Services");
			lblListOfServices.setHorizontalAlignment(SwingConstants.CENTER);
			lblListOfServices.setForeground(Color.WHITE);
			lblListOfServices.setFont(new Font("Bahnschrift", Font.PLAIN, 26));
			lblListOfServices.setBounds(405, 0, 609, 30);
			panel.add(lblListOfServices);

			JButton button_1 = new JButton("Back");
			button_1.addActionListener(new ActionListener() {
				public void actionPerformed(ActionEvent e) {
					dispose();
				}
			});
			button_1.setForeground(Color.BLACK);
			button_1.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
			button_1.setBackground(Color.WHITE);
			button_1.setBounds(1222, 5, 91, 25);
			panel.add(button_1);

		}
}
```

15. EditBranch.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
```

```java
import java.sql.ResultSet;
import java.sql.Statement;
import java.util.Random;

import javax.swing.JButton;
import javax.swing.JFormattedTextField;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JTextField;
import javax.swing.border.EmptyBorder;

public class EditBranch extends JFrame {

    private JPanel contentPane;
    private JTextField branchID;
    String address, con, id, query, check;
    Connection con1;
    String branch = "BRANCH";
    PreparedStatement pst, pst1;
    Random r = new Random();
    ResultSet rs;
    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    EditBranch frame = new EditBranch();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public EditBranch() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 868, 220);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setLayout(null);
        panel.setBorder(new EmptyBorder(5, 5, 5, 5));
        panel.setBounds(0, 0, 854, 183);
        contentPane.add(panel);

        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBackground(new Color(240, 255, 255));
```

```
panel_1.setBounds(0, 0, 862, 36);
panel.add(panel_1);

JLabel label = new JLabel("Edit Branch");
label.setForeground(new Color(0, 124, 124));
label.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
label.setBounds(0, 0, 882, 36);
panel_1.add(label);

JPanel panel_2 = new JPanel();
panel_2.setLayout(null);
panel_2.setBackground(Color.DARK_GRAY);
panel_2.setBounds(0, 31, 863, 158);
panel.add(panel_2);

JLabel label_1 = new JLabel("");
label_1.setForeground(Color.RED);
label_1.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
label_1.setBounds(445, 87, 176, 23);
panel_2.add(label_1);

JLabel label_2 = new JLabel("");
label_2.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
label_2.setBounds(0, 0, 0, 0);
panel_2.add(label_2);

branchID = new JTextField();
branchID.setForeground(Color.BLACK);
branchID.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
branchID.setColumns(10);
branchID.setBackground(Color.WHITE);
branchID.setBounds(152, 23, 469, 23);
panel_2.add(branchID);

JLabel label_3 = new JLabel("Branch ID:");
label_3.setForeground(Color.WHITE);
label_3.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
label_3.setBounds(10, 21, 132, 23);
panel_2.add(label_3);

JLabel label_4 = new JLabel("Address:");
label_4.setForeground(Color.WHITE);
label_4.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
label_4.setBounds(10, 54, 132, 23);
panel_2.add(label_4);

JLabel label_5 = new JLabel("Contact Number:");
label_5.setForeground(Color.WHITE);
label_5.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
label_5.setBounds(10, 87, 132, 23);
panel_2.add(label_5);

JFormattedTextField branchAddress = new JFormattedTextField();
branchAddress.setBounds(152, 56, 698, 23);
panel_2.add(branchAddress);

JLabel label_6 = new JLabel("");
label_6.setForeground(Color.RED);
label_6.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
```

```
        label_6.setBounds(445, 87, 176, 23);
        panel_2.add(label_6);

        JFormattedTextField branchContact = new JFormattedTextField();
        branchContact.setBounds(152, 87, 283, 23);
        panel_2.add(branchContact);

        JButton button = new JButton("Edit Branch");
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                address = branchAddress.getText();
                con = branchContact.getText();
                id = branchID.getText();
                query = "update branchlist set Address='" + address +
"', ContactNumber='" + con + "' where BranchID='" + id +"'";
                check = "SELECT * FROM branchlist";
                try {
                    con1 =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                    pst = con1.prepareStatement(query);
                    pst1 = con1.prepareStatement(check);
                    rs = pst1.executeQuery(check);
                    if (rs.next() == true) {
                        pst.execute();
                        JOptionPane.showMessageDialog(null,
"Branch found and edited successfully");
                        branchID.setText("");
                        branchAddress.setText("");
                        branchContact.setText("");
                    }
                    else if (branchID.getText().isEmpty() == true ||
branchAddress.getText().isEmpty() == true || branchContact.getText().isEmpty() ==
true) {
                        JOptionPane.showMessageDialog(null, "Some
entries are empty, re-enter");
                    }
                    else {
                        JOptionPane.showMessageDialog(null,
"Branch not found");
                    }
                }
                catch(Exception e1) {
                    JOptionPane.showMessageDialog(null, e1);
                }
            }
        });
        button.setForeground(Color.BLACK);
        button.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button.setBackground(Color.WHITE);
        button.setBounds(152, 120, 167, 23);
        panel_2.add(button);

        JButton button_1 = new JButton("Clear Contents");
        button_1.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                branchID.setText("");
                branchAddress.setText("");
                branchContact.setText("");
```

```
			}
		});
		button_1.setForeground(Color.BLACK);
		button_1.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
		button_1.setBackground(Color.WHITE);
		button_1.setBounds(329, 120, 167, 23);
		panel_2.add(button_1);

		JButton button_2 = new JButton("Back");
		button_2.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				dispose();
			}
		});
		button_2.setForeground(Color.BLACK);
		button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
		button_2.setBackground(Color.WHITE);
		button_2.setBounds(506, 120, 167, 23);
		panel_2.add(button_2);
	}
}
```

16. EditClient.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.KeyAdapter;
import java.awt.event.KeyEvent;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.text.SimpleDateFormat;
import java.util.Date;

import javax.swing.ButtonGroup;
import javax.swing.JButton;
import javax.swing.JFormattedTextField;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JRadioButton;
import javax.swing.JTextField;
import javax.swing.border.EmptyBorder;

import com.toedter.calendar.JDateChooser;

public class EditClient extends JFrame {

	private JPanel contentPane;
	private JTextField clientID;
	JDateChooser bdayChoose;
	JLabel label_1, label_2, label_3, label_4, label_5, label_6, label_7, 
label_8, lblNewLabel_3;
	JButton button_1, button_2, btnEditClient;
```

```java
JFormattedTextField clientContact, clientName, clientEmail;
Connection con, con1;
char vchar, gender1;
String id1, name1, email1, number1, date;
Date date1;
Function f;
ResultSet rs;
/**
 * Launch the application.
 */
public static void main(String[] args) {
    EventQueue.invokeLater(new Runnable() {
        public void run() {
            try {
                EditClient frame = new EditClient();
                frame.setVisible(true);
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    });
}

/**
 * Create the frame.
 */

String id, name, number, email, bday;
String gender;
PreparedStatement pst, pst1;

public void clearContents() {
    clientID.setText("");
    clientContact.setText("");
    clientName.setText("");
    clientEmail.setText("");
    bdayChoose.setDate(null);
}

public void textFieldEditable(boolean n) {
    clientName.setEditable(n);
    clientContact.setEditable(n);
    clientEmail.setEditable(n);
    bdayChoose.setEnabled(n);
}

public EditClient() {
    setBounds(100, 300, 706, 286);
    contentPane = new JPanel();
    contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
    setContentPane(contentPane);
    contentPane.setLayout(null);
    contentPane.setLayout(null);

    JPanel panel = new JPanel();
    panel.setLayout(null);
    panel.setBackground(new Color(240, 255, 255));
    panel.setBounds(0, 0, 863, 36);
    contentPane.add(panel);
```

```java
JLabel lblEditAnExisting = new JLabel("Edit an existing Client");
lblEditAnExisting.setForeground(new Color(0, 124, 124));
lblEditAnExisting.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
lblEditAnExisting.setBounds(0, 0, 692, 26);
panel.add(lblEditAnExisting);

JPanel panel_1 = new JPanel();
panel_1.setLayout(null);
panel_1.setBackground(Color.DARK_GRAY);
panel_1.setBounds(0, 31, 692, 218);
contentPane.add(panel_1);

label_1 = new JLabel("");
label_1.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
label_1.setBounds(0, 0, 0, 0);
panel_1.add(label_1);

clientID = new JTextField();
clientID.setForeground(Color.BLACK);
clientID.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
clientID.setColumns(10);
clientID.setBackground(Color.WHITE);
clientID.setBounds(152, 22, 283, 23);
panel_1.add(clientID);

JRadioButton radioButton = new JRadioButton("Male");
radioButton.setForeground(Color.WHITE);
radioButton.setFont(new Font("Bahnschrift", Font.BOLD, 12));
radioButton.setBackground(Color.DARK_GRAY);
radioButton.setBounds(152, 154, 51, 21);
panel_1.add(radioButton);

JRadioButton radioButton_1 = new JRadioButton("Female");
radioButton_1.setForeground(Color.WHITE);
radioButton_1.setFont(new Font("Bahnschrift", Font.BOLD, 12));
radioButton_1.setBackground(Color.DARK_GRAY);
radioButton_1.setBounds(205, 154, 80, 21);
panel_1.add(radioButton_1);

ButtonGroup bg = new ButtonGroup();
bg.add(radioButton);
bg.add(radioButton_1);

label_2 = new JLabel("Client ID:");
label_2.setForeground(Color.WHITE);
label_2.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
label_2.setBounds(10, 23, 132, 23);
panel_1.add(label_2);

label_3 = new JLabel("Mobile Number:");
label_3.setForeground(Color.WHITE);
label_3.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
label_3.setBounds(10, 85, 132, 23);
panel_1.add(label_3);

label_4 = new JLabel("");
label_4.setForeground(Color.RED);
label_4.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
```

```
            label_4.setBounds(445, 87, 176, 23);
            panel_1.add(label_4);

            lblNewLabel_3 = new JLabel("");
            lblNewLabel_3.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
            lblNewLabel_3.setForeground(Color.RED);
            lblNewLabel_3.setBounds(445, 87, 176, 23);
            panel_1.add(lblNewLabel_3);

            clientContact = new JFormattedTextField();
            clientContact.addKeyListener(new KeyAdapter() {
                @Override
                public void keyTyped(KeyEvent e) {
                    vchar = e.getKeyChar();
                    if ((!(Character.isDigit(vchar)) ||  (vchar ==
KeyEvent.VK_BACK_SPACE) ||  (vchar == KeyEvent.VK_DELETE))) {
                        lblNewLabel_3.setText("Invalid Entry!");
                        e.consume();
                    }
                    else {
                        lblNewLabel_3.setText("");
                    }
                }
            });
            clientContact.setBounds(152, 87, 283, 23);
            panel_1.add(clientContact);

            clientName = new JFormattedTextField();
            clientName.setBounds(152, 56, 283, 23);
            panel_1.add(clientName);

            clientEmail = new JFormattedTextField();
            clientEmail.setBounds(152, 120, 522, 23);
            panel_1.add(clientEmail);

            bdayChoose = new JDateChooser();
            bdayChoose.setBounds(371, 151, 250, 23);
            panel_1.add(bdayChoose);

            textFieldEditable(false);

            btnEditClient = new JButton("Edit Client");
            btnEditClient.setVisible(false);
            btnEditClient.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                    id1 = clientID.getText();
                    name1 = clientName.getText();
                    number1 = clientContact.getText();
                    email1 = clientEmail.getText().toLowerCase();
                    gender1 = 'M';
                    SimpleDateFormat sdf = new
SimpleDateFormat("yyyy/MM/dd");
                    bday = sdf.format(bdayChoose.getDate());
                    if (radioButton.isSelected()) {
                        gender1 = 'M';
                    }
                    else if (radioButton_1.isSelected()){
                        gender1 = 'F';
                    }
```

```
                          String query = "update clientlist set Name = '" + name1
+"', MobileNumber = '" + number1 +"',"
                                        + " Email = '" + email1 +"', Gender = '" +
gender1 + "', Birthday = '" + bday +"' "
                                                     + "where ClientID = '" + id1
+"'";
                          try {
                                 con1 =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                                 PreparedStatement pst1 =
con1.prepareStatement(query);
                                 if (name1.isEmpty()==false &&
number1.isEmpty()==false && email1.isEmpty()==false && bday.isEmpty()==false
                                              && (radioButton.isSelected()==true
|| radioButton_1.isSelected()==true)){
                                        pst1.execute();
                                        JOptionPane.showMessageDialog(null,
"Client Edited Successfully!");
                                        textFieldEditable(false);
                                        clientID.setEditable(true);
                                        clearContents();
                                 }
                                 else {
                                        JOptionPane.showMessageDialog(null, "One
or more fields are empty, please fill all the fields!");
                                 }
                          }
                          catch(Exception e1) {
                                 JOptionPane.showMessageDialog(null, e1);
                          }
                    }
             });
             btnEditClient.setForeground(Color.BLACK);
             btnEditClient.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
             btnEditClient.setBackground(Color.WHITE);
             btnEditClient.setBounds(152, 181, 167, 23);
             panel_1.add(btnEditClient);

             button_1 = new JButton("Clear Contents");
             button_1.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                          clearContents();
                    }
             });
             button_1.setForeground(Color.BLACK);
             button_1.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
             button_1.setBackground(Color.WHITE);
             button_1.setBounds(329, 181, 167, 23);
             panel_1.add(button_1);

             button_2 = new JButton("Back");
             button_2.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                          dispose();
                    }
             });
             button_2.setForeground(Color.BLACK);
             button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
```

```
        button_2.setBackground(Color.WHITE);
        button_2.setBounds(506, 181, 167, 23);
        panel_1.add(button_2);

        label_5 = new JLabel("Email:");
        label_5.setForeground(Color.WHITE);
        label_5.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        label_5.setBounds(10, 116, 132, 23);
        panel_1.add(label_5);

        label_6 = new JLabel("Gender:");
        label_6.setForeground(Color.WHITE);
        label_6.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        label_6.setBounds(10, 147, 132, 23);
        panel_1.add(label_6);

        label_7 = new JLabel("Birthdate:");
        label_7.setForeground(Color.WHITE);
        label_7.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        label_7.setBounds(291, 153, 85, 23);
        panel_1.add(label_7);

        label_8 = new JLabel("Name");
        label_8.setForeground(Color.WHITE);
        label_8.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        label_8.setBounds(10, 54, 132, 23);
        panel_1.add(label_8);

        JButton btnSearch = new JButton("Search");
        btnSearch.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                f = new Function();
                rs = null;
                name = "Name";
                number = "MobileNumber";
                email = "Email";
                gender = "Gender";
                date = "Birthday";
                if (clientID.getText().isEmpty() == true) {
                    JOptionPane.showMessageDialog(null, "Field Blank!
Please enter a Client ID!");
                }
                else {
                    rs = f.find(clientID.getText());
                    try {
                        if (rs.next()) {
                            textFieldEditable(true);

    clientName.setText(rs.getString(name));

    clientContact.setText(rs.getString(number));

    clientEmail.setText(rs.getString(email));
                            if
(rs.getString(gender).equals("M")) {

    radioButton.setSelected(true);
                            }
                            else
if(rs.getString(gender).equals("F"))
                            {

    radioButton_1.setSelected(true);
                            }
                            bday = rs.getString(date);
                            date1 = new
SimpleDateFormat("yyyy/MM/dd").parse(bday);
                            bdayChoose.setDate(date1);
                            btnEditClient.setVisible(true);
                            clientID.setEditable(false);
                        }
                    }
                    catch(Exception e1) {
                        JOptionPane.showMessageDialog(null, e1);
                    }
                }
            }
        });
        btnSearch.setForeground(Color.BLACK);
        btnSearch.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnSearch.setBackground(Color.WHITE);
        btnSearch.setBounds(445, 22, 167, 23);
        panel_1.add(btnSearch);
    }
}
```

17. EditProduct.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.KeyAdapter;
import java.awt.event.KeyEvent;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

import javax.swing.JButton;
import javax.swing.JFormattedTextField;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JTextArea;
import javax.swing.border.EmptyBorder;

public class EditProduct extends JFrame {

    private JPanel contentPane;
    JFormattedTextField prodComp, prodName, prodSize, prodRate, productID;
    JTextArea prodDesc;
    int srlno;
    String company1;
    String pname1;
    String description1;
    int size1;
    int rate1;
    String query;
    Connection con1;
    Function f;
    ResultSet rs;
    String company;
    String pname;
    String description;
    String size;
    String rate;
    PreparedStatement pst1;
    char vchar;
    /**
     * Launch the application.
     */
    public static void main(String[] args) {
            EventQueue.invokeLater(new Runnable() {
                    public void run() {
                            try {
                                    EditProduct frame = new EditProduct();
                                    frame.setVisible(true);
                            } catch (Exception e) {
                                    e.printStackTrace();
                            }
                    }
            });
    }

    /**
     * Create the frame.
     */



    public void clearContents() {
            prodDesc.setText("");
            prodComp.setText("");
            prodName.setText("");
            prodSize.setText("");
            prodRate.setText("");
            productID.setText("");
    }

    public void textFieldEditable(boolean n) {
            prodDesc.setEditable(n);
            prodComp.setEditable(n);
            prodName.setEditable(n);
            prodSize.setEditable(n);
            prodRate.setEditable(n);
    }
    public EditProduct() {
            setBounds(100, 300, 626, 303);
            contentPane = new JPanel();
            contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
            setContentPane(contentPane);
            contentPane.setLayout(null);
            JPanel panel = new JPanel();
            panel.setLayout(null);
            panel.setBackground(new Color(240, 255, 255));
            panel.setBounds(0, 0, 612, 28);
            contentPane.add(panel);

            JLabel lblEditAProduct = new JLabel("Edit a product");
            lblEditAProduct.setForeground(new Color(0, 124, 124));
            lblEditAProduct.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
            lblEditAProduct.setBounds(0, 0, 692, 26);
            panel.add(lblEditAProduct);

            JPanel panel_1 = new JPanel();
            panel_1.setLayout(null);
            panel_1.setBackground(Color.DARK_GRAY);
            panel_1.setBounds(0, 27, 612, 269);
            contentPane.add(panel_1);

            JLabel label_2 = new JLabel("Company:");
            label_2.setForeground(Color.WHITE);
            label_2.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
            label_2.setBounds(10, 38, 86, 23);
            panel_1.add(label_2);

            JLabel label_3 = new JLabel("Size (in ML):");
            label_3.setForeground(Color.WHITE);
            label_3.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
            label_3.setBounds(10, 102, 94, 23);
            panel_1.add(label_3);

            JButton btnEditProduct = new JButton("Edit Product");
            btnEditProduct.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                            srlno = Integer.parseInt(productID.getText());
                            company1 = prodComp.getText();
                            pname1 = prodName.getText();
                            description1 = prodDesc.getText();
                            size1 = Integer.parseInt(prodSize.getText());
                            rate1 = Integer.parseInt(prodRate.getText());
                            query = "UPDATE productlist SET Company = 
            '"+company1+"', ProductName = '"+pname1+"', ProductDescription = 
            '"+description1+"', Size = '"+size1+"', Rate = '"+rate1+"' WHERE (SerialNo = 
            '"+srlno+"')";
                            try {
                                con1 = 
            DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root", 
            "mySQLpassword");
                                pst1 = con1.prepareStatement(query);
                                if (company1.isEmpty()==false && 
            pname1.isEmpty()==false && prodSize.getText().isEmpty()==false && 
            prodRate.getText().isEmpty()==false) {
                                    pst1.execute();
                                    JOptionPane.showMessageDialog(null, 
            "Product edited successfully!");
                                    clearContents();
                                    textFieldEditable(false);
                                    productID.setEditable(true);
                                }
                                else {
                                    JOptionPane.showMessageDialog(null, "One
            or more fields are empty, please fill all the fields!");
                                }
                            }
                            catch(Exception e1) {
                                JOptionPane.showMessageDialog(null, e1);
                            }
                    }
            });
            btnEditProduct.setForeground(Color.BLACK);
            btnEditProduct.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
            btnEditProduct.setBackground(Color.WHITE);
            btnEditProduct.setBounds(121, 196, 167, 23);
            panel_1.add(btnEditProduct);

            JButton button_1 = new JButton("Clear Contents");
            button_1.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                            clearContents();
                    }
            });
            button_1.setForeground(Color.BLACK);
            button_1.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
            button_1.setBackground(Color.WHITE);
            button_1.setBounds(298, 196, 167, 23);
            panel_1.add(button_1);

            JButton button_2 = new JButton("Back");
            button_2.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                            dispose();
                    }
            });
            button_2.setForeground(Color.BLACK);
            button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
            button_2.setBackground(Color.WHITE);
            button_2.setBounds(475, 196, 122, 23);
            panel_1.add(button_2);

            JLabel label_5 = new JLabel("Rate (in \u20B9):");
            label_5.setForeground(Color.WHITE);
            label_5.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
            label_5.setBounds(336, 102, 93, 23);
            panel_1.add(label_5);

            JLabel lblProductName = new JLabel("Product Name:");
            lblProductName.setForeground(Color.WHITE);
            lblProductName.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
            lblProductName.setBounds(10, 69, 109, 23);
            panel_1.add(lblProductName);

            JLabel label_8 = new JLabel("<HTML>Product Description:<HTML>");
            label_8.setForeground(Color.WHITE);
            label_8.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
            label_8.setBounds(10, 139, 86, 51);
            panel_1.add(label_8);

            prodComp = new JFormattedTextField();
            prodComp.setBounds(121, 40, 476, 23);
            panel_1.add(prodComp);

            prodName = new JFormattedTextField();
            prodName.setBounds(121, 71, 476, 23);
            panel_1.add(prodName);

            JLabel lblNewLabel_3 = new JLabel("");
            lblNewLabel_3.setForeground(Color.RED);
            lblNewLabel_3.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
            lblNewLabel_3.setBounds(10, 120, 104, 23);
            panel_1.add(lblNewLabel_3);

            prodSize = new JFormattedTextField();
            prodSize.addKeyListener(new KeyAdapter() {
                @Override
                public void keyTyped(KeyEvent e) {
                         vchar = e.getKeyChar();
                         if ((!(Character.isDigit(vchar)) ||  (vchar ==
         KeyEvent.VK_BACK_SPACE) ||  (vchar == KeyEvent.VK_DELETE))) {
                             lblNewLabel_3.setText("Invalid Entry!");
                             e.consume();
                         }
                         else {
                             lblNewLabel_3.setText("");
                         }
                }
            });
            prodSize.setBounds(121, 102, 172, 23);
            panel_1.add(prodSize);

            prodRate = new JFormattedTextField();
            prodRate.addKeyListener(new KeyAdapter() {
                @Override
                public void keyTyped(KeyEvent e) {
                         vchar = e.getKeyChar();
                         if ((!(Character.isDigit(vchar)) ||  (vchar ==
         KeyEvent.VK_BACK_SPACE) ||  (vchar == KeyEvent.VK_DELETE))) {
                             lblNewLabel_3.setText("Invalid Entry!");
                             e.consume();
                         }
                         else {
                             lblNewLabel_3.setText("");
                         }
                }
            });
            prodRate.setBounds(425, 102, 172, 23);
            panel_1.add(prodRate);

            prodDesc = new JTextArea();
            prodDesc.setLineWrap(true);
            prodDesc.setBounds(121, 139, 476, 51);
            panel_1.add(prodDesc);

            JLabel lblSerialNumber = new JLabel("Serial Number:");
            lblSerialNumber.setForeground(Color.WHITE);
            lblSerialNumber.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
            lblSerialNumber.setBounds(10, 5, 122, 23);
            panel_1.add(lblSerialNumber);
            textFieldEditable(false);

            productID = new JFormattedTextField();
            productID.setBounds(138, 5, 150, 23);
            panel_1.add(productID);

            JButton btnSearch = new JButton("Search");
            btnSearch.addActionListener(new ActionListener() {
                public void actionPerformed(ActionEvent e) {
                         f = new Function();
                         rs = null;
                         company = "Company";
                         pname = "ProductName";
                         description = "ProductDescription";
                         size = "Size";
                         rate = "Rate";
                         if(productID.getText().isEmpty() == true) {
                             JOptionPane.showMessageDialog(null, "Field Blank!
         Please enter a Product Serial No.!");
                         }
                         else {
                             rs=f.find(Integer.parseInt(productID.getText()));
                             try {
                                 if (rs.next()) {
                                     textFieldEditable(true);

             prodComp.setText(rs.getString(company));

             prodName.setText(rs.getString(pname));

             prodDesc.setText(rs.getString(description));

             prodSize.setText(Integer.toString(rs.getInt(size)));

             prodRate.setText(Integer.toString(rs.getInt(rate)));
                                     productID.setEditable(false);
                                 }
                                 else {
                                     JOptionPane.showMessageDialog(null,
         "No product registered with that serial number");
                                 }
                             }
                             catch(Exception e1) {

                             }
                         }
                }
            });
            btnSearch.setForeground(Color.BLACK);
            btnSearch.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
            btnSearch.setBackground(Color.WHITE);
            btnSearch.setBounds(298, 5, 109, 23);
            panel_1.add(btnSearch);
             }
         }
```

18. EditService.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

import javax.swing.JButton;
import javax.swing.JFormattedTextField;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;

public class EditService extends JFrame {

    private JPanel contentPane;
    JFormattedTextField serviceCat, serviceSubCat, serviceRate, serviceSrlNo;
    int srlno, rate;
    String cat, subcat, query;
    Connection con1;
    PreparedStatement pst1;
    Function f;
    ResultSet rs;
    String rate1;
    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    EditService frame = new EditService();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */



    public void textFieldEditable(boolean n) {
        serviceCat.setEditable(n);
        serviceSubCat.setEditable(n);
        serviceRate.setEditable(n);
    }

    public void clearContents() {
        textFieldEditable(false);
        serviceSrlNo.setEditable(true);
        serviceCat.setText("");
        serviceSubCat.setText("");
        serviceRate.setText("");
        serviceSrlNo.setText("");
    }

    public EditService() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 553, 237);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setLayout(null);
        panel.setBackground(new Color(240, 255, 255));
        panel.setBounds(0, 0, 612, 28);
        contentPane.add(panel);

        JLabel label = new JLabel("Edit an existing Service");
        label.setForeground(new Color(0, 124, 124));
        label.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
        label.setBounds(0, 0, 537, 28);
        panel.add(label);

        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBackground(Color.DARK_GRAY);
        panel_1.setBounds(0, 30, 539, 186);
        contentPane.add(panel_1);

        JLabel label_1 = new JLabel("Category:");
        label_1.setForeground(Color.WHITE);
        label_1.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        label_1.setBounds(10, 38, 86, 23);
        panel_1.add(label_1);

        JButton button = new JButton("Edit Service");
        button.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                srlno = Integer.parseInt(serviceSrlNo.getText());
                cat = serviceCat.getText();
                subcat = serviceSubCat.getText();
                rate = Integer.parseInt(serviceRate.getText());
                query = "UPDATE servicelist SET Category = '"+cat+"', 
Subcategory = '"+subcat+"', Rate = '"+rate+"' WHERE (SerialNo = '"+srlno+"')";
                try {
                    con1 = 
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root", 
"mySQLpassword");
                    pst1 = con1.prepareStatement(query);
                    if (serviceCat.getText().isEmpty()==false && 
serviceSubCat.getText().isEmpty()==false && serviceCat.getText().isEmpty()==false) 
{
                        pst1.execute();
                        JOptionPane.showMessageDialog(null, 
"Service edited successfully!");
                        clearContents();
```

```
                                textFieldEditable(false);
                                serviceSrlNo.setEditable(true);
                            }
                            else {
                                JOptionPane.showMessageDialog(null, "One
        or more fields are empty, please fill all the fields!");
                            }
                        }
                        catch(Exception e1) {
                            JOptionPane.showMessageDialog(null, e1);
                        }
                    }
                });
                button.setForeground(Color.BLACK);
                button.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
                button.setBackground(Color.WHITE);
                button.setBounds(121, 135, 153, 23);
                panel_1.add(button);

                JButton button_1 = new JButton("Clear Contents");
                button_1.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                        clearContents();
                    }
                });
                button_1.setForeground(Color.BLACK);
                button_1.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
                button_1.setBackground(Color.WHITE);
                button_1.setBounds(284, 135, 132, 23);
                panel_1.add(button_1);

                JButton button_2 = new JButton("Back");
                button_2.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                        dispose();
                    }
                });
                button_2.setForeground(Color.BLACK);
                button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
                button_2.setBackground(Color.WHITE);
                button_2.setBounds(426, 135, 74, 23);
                panel_1.add(button_2);

                JLabel label_2 = new JLabel("Rate (in \u20B9):");
                label_2.setForeground(Color.WHITE);
                label_2.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
                label_2.setBounds(10, 102, 93, 23);
                panel_1.add(label_2);

                JLabel label_3 = new JLabel("Sub-category:");
                label_3.setForeground(Color.WHITE);
                label_3.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
                label_3.setBounds(10, 69, 132, 23);
                panel_1.add(label_3);

                serviceCat = new JFormattedTextField();
                serviceCat.setBounds(121, 40, 379, 23);
                panel_1.add(serviceCat);
                serviceSubCat = new JFormattedTextField();
                serviceSubCat.setBounds(121, 71, 379, 23);
                panel_1.add(serviceSubCat);

                JLabel label_4 = new JLabel("");
                label_4.setForeground(Color.RED);
                label_4.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
                label_4.setBounds(217, 102, 104, 23);
                panel_1.add(label_4);

                serviceRate = new JFormattedTextField();
                serviceRate.setBounds(121, 102, 86, 23);
                panel_1.add(serviceRate);

                textFieldEditable(false);

                JLabel lblEnterSerialNumber = new JLabel("Serial Number:");
                lblEnterSerialNumber.setForeground(Color.WHITE);
                lblEnterSerialNumber.setFont(new Font("Bahnschrift", Font.PLAIN,
   16));
                lblEnterSerialNumber.setBounds(10, 10, 110, 23);
                panel_1.add(lblEnterSerialNumber);

                serviceSrlNo = new JFormattedTextField();
                serviceSrlNo.setBounds(121, 7, 153, 23);
                panel_1.add(serviceSrlNo);

                JButton btnSearch = new JButton("Search");
                btnSearch.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                             f = new Function();
                             rs = null;
                             cat = "Category";
                             subcat = "Subcategory";
                             rate1 = "Rate";
                             if(serviceSrlNo.getText().isEmpty() == true) {
                                    JOptionPane.showMessageDialog(null, "Field Blank!
   Please enter the Service Serial No.!");
                             }
                             else {

                 rs=f.find1(Integer.parseInt(serviceSrlNo.getText()));
                                    try {
                                        if (rs.next()) {
                                            textFieldEditable(true);

                 serviceCat.setText(rs.getString(cat));

                 serviceSubCat.setText(rs.getString(subcat));

                 serviceRate.setText(Integer.toString(rs.getInt(rate1)));
                                            serviceSrlNo.setEditable(false);
                                        }
                                        else {
                                            JOptionPane.showMessageDialog(null,
   "No service registered with that serial number");
                                        }
                                    }
                                    catch(Exception e1) {
                                    }
                             }
                    }
                });
                btnSearch.setForeground(Color.BLACK);
                btnSearch.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
                btnSearch.setBackground(Color.WHITE);
                btnSearch.setBounds(282, 7, 153, 23);
                panel_1.add(btnSearch);
                 }

   }
```

19. Function.java

```
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;

import javax.swing.JOptionPane;

public class Function{
        Connection con = null;
        ResultSet rs = null;
        PreparedStatement pst = null;
        public ResultSet find(int s) {
            try {
            con = 
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root", 
"mySQLpassword");
            pst = con.prepareStatement("SELECT * FROM productlist WHERE 
SerialNo = ?");
            pst.setInt(1, s);
            rs=pst.executeQuery();
            }
            catch(Exception e) {
                JOptionPane.showMessageDialog(null, e.getMessage());
            }
            return rs;
        }

        public ResultSet find1(int v) {
            try {
            con = 
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root", 
"mySQLpassword");
            pst = con.prepareStatement("SELECT * FROM servicelist WHERE 
SerialNo = ?");
            pst.setInt(1, v);
            rs=pst.executeQuery();
            }
            catch(Exception e) {
                JOptionPane.showMessageDialog(null, e.getMessage());
            }
            return rs;
        }
        public ResultSet find(String s) {
            try {
            con = 
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root", 
"mySQLpassword");
            pst = con.prepareStatement("SELECT * FROM clientlist WHERE 
ClientID = ?");
            pst.setString(1, s);
            rs=pst.executeQuery();
            }
            catch(Exception e) {
                JOptionPane.showMessageDialog(null, e.getMessage());
            }
            return rs;
        }

    }
```

20. GenRep.java

```
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 javax.swing.SwingConstants;
import java.awt.Color;
import java.awt.Font;
import java.awt.Window;

import com.itextpdf.text.BaseColor;
import com.itextpdf.text.Document;
import com.itextpdf.text.Element;
import com.itextpdf.text.FontFactory;
import com.itextpdf.text.Paragraph;
import com.itextpdf.text.Phrase;
import com.itextpdf.text.pdf.BaseFont;
import com.itextpdf.text.pdf.PdfPCell;
import com.itextpdf.text.pdf.PdfPTable;
import com.itextpdf.text.pdf.PdfWriter;
import com.toedter.calendar.JDateChooser;

import net.proteanit.sql.DbUtils;

import javax.swing.ButtonGroup;
import javax.swing.JButton;
import javax.swing.JFileChooser;

import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.Date;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
```

```java
import java.util.LinkedList;
import java.awt.event.ActionEvent;
import javax.swing.JTable;
import javax.swing.JScrollPane;
import javax.swing.table.DefaultTableModel;
import javax.swing.table.TableColumnModel;
import javax.swing.JToggleButton;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.io.FileOutputStream;

public class GenRep extends JFrame {

        private JPanel contentPane;
        private JTable table;
        private JTable table_6;
        java.sql.Date date1, date2;
        JButton btnGenerateAPdf;
        String sum;
        int tot = 0;
        String impQuery, impQuery1, query, query1, query2, query11;
        Connection con;
        int val1, val2;
        PreparedStatement pst, pstDate, pst1, pst2;
        Date lastVisit;
        ResultSet rs, rsDate, rs1, rs2;
        int g;
        String path1, path;
        JFileChooser j;
        java.util.Date myDateObj;
        SimpleDateFormat formatter;
    java.sql.Date formattedDate;

        /**
         * Launch the application.
         */
        public static void main(String[] args) {
                EventQueue.invokeLater(new Runnable() {
                        public void run() {
                                try {
                                        GenRep frame = new GenRep();
                                        frame.setVisible(true);
                                } catch (Exception e) {
                                        e.printStackTrace();
                                }
                        }
                });
        }

        /**
         * Create the frame.
         */
        public GenRep() {
                setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
                setBounds(100, 100, 635, 454);
                contentPane = new JPanel();
                contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
                setContentPane(contentPane);
                contentPane.setLayout(null);
```

```
JPanel panel = new JPanel();
panel.setBounds(0, 0, 621, 425);
panel.setBackground(Color.CYAN.darker().darker().darker());
contentPane.add(panel);
panel.setLayout(null);

ButtonGroup bg = new ButtonGroup();

ButtonGroup bg1 = new ButtonGroup();

JLabel lblGenerateReports = new JLabel("GENERATE REPORT");
lblGenerateReports.setFont(new Font("Bahnschrift", Font.PLAIN, 36));
lblGenerateReports.setForeground(Color.WHITE);
lblGenerateReports.setHorizontalAlignment(SwingConstants.CENTER);
lblGenerateReports.setBounds(0, 0, 601, 66);
panel.add(lblGenerateReports);

JLabel lblNumberOfClients = new JLabel("Number of clients:");
lblNumberOfClients.setForeground(Color.WHITE);
lblNumberOfClients.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
lblNumberOfClients.setBounds(10, 103, 394, 19);
panel.add(lblNumberOfClients);

JLabel noOfClients = new JLabel("");
noOfClients.setForeground(Color.GREEN);
noOfClients.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
noOfClients.setBounds(414, 103, 101, 19);
panel.add(noOfClients);

JLabel lblAmountOfSales = new JLabel("Amount of sales:");
lblAmountOfSales.setHorizontalAlignment(SwingConstants.LEFT);
lblAmountOfSales.setForeground(Color.WHITE);
lblAmountOfSales.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
lblAmountOfSales.setBounds(10, 132, 144, 19);
panel.add(lblAmountOfSales);

JLabel sales = new JLabel("");
sales.setForeground(Color.GREEN);
sales.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
sales.setBounds(163, 132, 186, 19);
panel.add(sales);

JDateChooser dateChooser = new JDateChooser();
dateChooser.setBounds(96, 74, 172, 19);
panel.add(dateChooser);

JLabel lblFrom = new JLabel("From");
lblFrom.setForeground(Color.WHITE);
lblFrom.setFont(new Font("Bahnschrift", Font.PLAIN, 22));
lblFrom.setBounds(32, 76, 54, 19);
panel.add(lblFrom);

JLabel lblTo = new JLabel("To");
lblTo.setForeground(Color.WHITE);
lblTo.setFont(new Font("Bahnschrift", Font.PLAIN, 22));
lblTo.setBounds(278, 74, 29, 19);
panel.add(lblTo);
```

```
             JDateChooser dateChooser_1 = new JDateChooser();
             dateChooser_1.setBounds(308, 74, 172, 19);
             panel.add(dateChooser_1);

             JScrollPane scrollPane_1 = new JScrollPane();
             scrollPane_1.setBounds(10, 216, 296, 165);
             panel.add(scrollPane_1);

             table = new JTable();
             table.setAutoCreateRowSorter(true);
             scrollPane_1.setViewportView(table);

             JScrollPane scrollPane = new JScrollPane();
             scrollPane.setBounds(317, 216, 296, 165);
             panel.add(scrollPane);

             table_6 = new JTable();
             table_6.setAutoCreateRowSorter(true);
             scrollPane.setViewportView(table_6);

             JButton btnGenerate = new JButton("Generate");
             btnGenerate.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                          try {
                                 btnGenerateAPdf.setVisible(true);
                                 date1 = new
java.sql.Date(dateChooser.getDate().getTime());
                                 date2 = new
java.sql.Date(dateChooser_1.getDate().getTime());

                                 try {
                                        impQuery = "select `Product Name`,
count(productID) AS `Products Bought` from invoiceproductlist where Date between
'"+date1+"' and '"+date2+"' GROUP BY productID ORDER BY `Products Bought` DESC";
                                        con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                                        pst = con.prepareStatement(impQuery);
                                        rs = pst.executeQuery();

       table.setModel(DbUtils.resultSetToTableModel(rs));
                                 }
                                 catch(Exception e1) {
                                        JOptionPane.showMessageDialog(null,
"Error");
                                 }

                                 try {
                                        impQuery1 = "select `Service Name`,
count(serviceID) AS `Services Taken` from invoiceservicelist where Date between
'"+date1+"' and '"+date2+"' GROUP BY serviceID order by `Services Taken` DESC";
                                        con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                                        pst = con.prepareStatement(impQuery1);
                                        rs = pst.executeQuery();

       table_6.setModel(DbUtils.resultSetToTableModel(rs));
                                 }
                    catch(Exception e1) {
                        JOptionPane.showMessageDialog(null,
"Error");
                    }

                    query = "select sum(Expense) from invoicelist
where Date between '"+date1+"' and '"+date2+"'";
                    con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                    pst = con.prepareStatement(query);
                    rs = pst.executeQuery();
                    lblNumberOfClients.setText("Number of clients
since " + dateChooser.getDate() + ":");
                    val1=0; val2 = 0;
                    query1 = "select SerialNo from invoicelist where
Date = '"+date1+"'";
                    query2 = "select SerialNo from invoicelist where
Date = '"+date2+"'";
                    query11 = "select Date from invoicelist";
                    pstDate = con.prepareStatement(query11);
                    rsDate = pstDate.executeQuery();
                    lastVisit = null;
                    while (rsDate.next()) {
                        lastVisit = rsDate.getDate("Date");
                    }
                    if (date2.compareTo(lastVisit) < 0 ||
date2.compareTo(lastVisit) == 0 || date2.compareTo(lastVisit) > 0) {
                        pst1 = con.prepareStatement(query1);
                        rs1 = pst1.executeQuery();
                        pst2 = con.prepareStatement(query2);
                        rs2 = pst2.executeQuery();
                        while (rs1.next()) {
                            val1 = rs1.getInt("SerialNo");
                        }
                        while (rs2.next()) {
                            val2 = rs2.getInt("SerialNo");
                        }
                        if (val1 > val2) {
                            tot = val1 - val2 + 1;
                        }
                        else if (val2 > val1) {
                            tot = val2 - val1 + 1;
                        }
                        noOfClients.setText(String.valueOf(tot));
                        rs.next();
                        sum = rs.getString(1);
                        sales.setText(sum);
                    }
                }
                catch(Exception e1) {
                    JOptionPane.showMessageDialog(null, "Error");
                }
            }
        });
        btnGenerate.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnGenerate.setBounds(490, 74, 98, 19);
        panel.add(btnGenerate);
```

```
             btnGenerateAPdf = new JButton("Generate a PDF Report");
             btnGenerateAPdf.setVisible(false);
             btnGenerateAPdf.addActionListener(new ActionListener() {
                   public void actionPerformed(ActionEvent e) {
                         path1="";
                         j = new JFileChooser();
                         j.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);
                         g = j.showSaveDialog(null);

                         if(g==JFileChooser.APPROVE_OPTION) {
                               path1 = j.getSelectedFile().toString();
                         }
                         int time = (int) (System.currentTimeMillis()/1000);
                         path1 = path1.replace("\\", "\\\\");
                         path = path1+"\\\\BM Report - "+time+".pdf";

                         myDateObj = new java.util.Date();
                         formatter = new SimpleDateFormat("dd/MM/yyyy
HH:mm:ss");
                       formattedDate = new java.sql.Date(myDateObj.getTime());

                         Document doc = new Document();
                         date1 = new
java.sql.Date(dateChooser.getDate().getTime());
                         date2 = new
java.sql.Date(dateChooser_1.getDate().getTime());
                         try{
                               String FONT1 = "resources/fonts/PlayfairDisplay-
Regular.ttf";
                               com.itextpdf.text.Font f3
=FontFactory.getFont(FONT1, BaseFont.IDENTITY_H, BaseFont.EMBEDDED, 18);
                               PdfWriter.getInstance(doc, new
FileOutputStream(path));

                               doc.open();

                               Paragraph para = new Paragraph("BEAUTYMANNTRA
INTERVAL REPORT",

      FontFactory.getFont(FontFactory.HELVETICA_BOLD, 24,
                                                          Font.BOLD,
BaseColor.CYAN.darker().darker().darker()));
                               para.setAlignment(Element.ALIGN_CENTER);
                               Paragraph para1 = new Paragraph("----------------
------------------------------------------"
                                             + "--------------------------------
---------------------------------------");
                               Paragraph para2 = new Paragraph("Report Generated
on: " + formatter.format(myDateObj),

      FontFactory.getFont(FontFactory.HELVETICA, 12,
                                                          Font.PLAIN,
BaseColor.DARK_GRAY));
                               para2.setAlignment(Element.ALIGN_RIGHT);
                               Paragraph paraSpace = new Paragraph("");
                               Paragraph space = new Paragraph("----------------
------------------------------------------"
                                             + "--------------------------------
---------------------------------------");
```

```
                                 Paragraph para3 = new Paragraph("Most to Least
Demanded Products between " + date1 + " and " + date2,

       FontFactory.getFont(FontFactory.HELVETICA, 15,
                                                           Font.PLAIN,
BaseColor.BLACK));
                                 para3.setAlignment(Element.ALIGN_CENTER);
                                 doc.add(para); doc.add(paraSpace);
doc.add(space); doc.add(paraSpace);
                                 doc.add(para1); doc.add(paraSpace);
doc.add(para2); doc.add(paraSpace);
                                 doc.add(space); doc.add(paraSpace);
doc.add(para3);

                                 PdfPTable table1 = new PdfPTable(3);
                                 table1.setWidthPercentage(105);
                                 table1.setSpacingBefore(11f);
                                 table1.setSpacingAfter(11f);

                                 float[] colWidth= {2f, 2f, 2f};
                                 table1.setWidths(colWidth);
                                 PdfPCell c1 = new PdfPCell(new Phrase("Serial
No"));

                                 table1.addCell(c1);
                                 PdfPCell c2 = new PdfPCell(new Phrase("Product
Name"));

                                 table1.addCell(c2);
                                 PdfPCell c3 = new PdfPCell(new Phrase("Number of
Products Bought"));

                                 table1.addCell(c3);

                                 for (int i = 0; i < table.getRowCount();i++) {
                                        String serialNo = String.valueOf(i+1);
                                        String prodName = table.getValueAt(i,
0).toString();

                                        String prodBought = table.getValueAt(i,
1).toString();

                                        table1.addCell(serialNo);
                                        table1.addCell(prodName);
                                        table1.addCell(prodBought);
                                 }
                                 doc.add(table1);
                                 doc.add(space);

                                 Paragraph paraServ = new Paragraph("Most to Least
Demanded Services between " + date1 + " and " + date2,

       FontFactory.getFont(FontFactory.HELVETICA, 15,
                                                           Font.PLAIN,
BaseColor.BLACK));
                                 para3.setAlignment(Element.ALIGN_CENTER);
                                 doc.add(paraServ);

                                 PdfPTable table2 = new PdfPTable(3);
                                 table2.setWidthPercentage(105);
                                 table2.setSpacingBefore(11f);
                                 table2.setSpacingAfter(11f);
                                 table2.setWidths(colWidth);
```

```
                    PdfPCell c11 = new PdfPCell(new Phrase("Serial
No"));
                    table2.addCell(c11);
                    PdfPCell c22 = new PdfPCell(new Phrase("Service
Name"));
                    table2.addCell(c22);
                    PdfPCell c33 = new PdfPCell(new Phrase("Frequency
of Services taken"));
                    table2.addCell(c33);

                    for (int i = 0; i < table_6.getRowCount();i++) {
                        String serialNo = String.valueOf(i+1);
                        String servName = table_6.getValueAt(i,
0).toString();
                        String servBought = table_6.getValueAt(i,
1).toString();
                        table2.addCell(serialNo);
                        table2.addCell(servName);
                        table2.addCell(servBought);
                    }
                    doc.add(table2);
                    doc.add(space);

                    Paragraph amountOfClients = new Paragraph("Amount
of Sales between " + date1 + " and " + date2 + " = " + sum,

    FontFactory.getFont(FontFactory.HELVETICA, 15,
                                    Font.PLAIN,
BaseColor.BLACK));

    amountOfClients.setAlignment(Element.ALIGN_CENTER);
                    doc.add(amountOfClients);

                    doc.add(space);

                    Paragraph amountOfSales = new Paragraph("Number
of Clients between " + date1 + " and " + date2 + " = " + tot,

    FontFactory.getFont(FontFactory.HELVETICA, 15,
                                    Font.PLAIN,
BaseColor.BLACK));
                    amountOfSales.setAlignment(Element.ALIGN_CENTER);
                    doc.add(amountOfSales);

                    doc.add(space);

                    Paragraph para5 = new Paragraph("Thank you for
using the program!");
                    para5.setAlignment(Element.ALIGN_CENTER);
                    doc.add(para5);
                    doc.add(space);
                    doc.close();
                }
                catch(Exception e1) {

                }
            }
        });
        btnGenerateAPdf.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
```

```
		btnGenerateAPdf.setBounds(118, 391, 189, 19);
		panel.add(btnGenerateAPdf);
		
		JButton btnBack = new JButton("Back");
		btnBack.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				dispose();
				Reports rp = new Reports();
				rp.setVisible(true);
			}
		});
		btnBack.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
		btnBack.setBounds(10, 391, 98, 19);
		panel.add(btnBack);
		
		JLabel lblProductsAndServices = new JLabel("Products and Services in 
most demand");
	
	lblProductsAndServices.setHorizontalAlignment(SwingConstants.CENTER);
		lblProductsAndServices.setForeground(Color.WHITE);
		lblProductsAndServices.setFont(new Font("Bahnschrift", Font.PLAIN, 
24));
		lblProductsAndServices.setBounds(0, 171, 621, 35);
		panel.add(lblProductsAndServices);
		
		JPanel panel_1 = new JPanel();
		panel_1.setBackground(Color.BLACK);
		panel_1.setBounds(0, 161, 621, 10);
		panel.add(panel_1);
	}
}
```

21. Invoices.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;

public class Invoices extends JFrame {

	private JPanel contentPane;

	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {
				try {
					Invoices frame = new Invoices();
```

```
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public Invoices() {
        setBounds(100, 100, 504, 181);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setLayout(null);
        panel.setBorder(new EmptyBorder(5, 5, 5, 5));
        panel.setBackground(new Color(240, 255, 255));
        panel.setBounds(0, 0, 977, 144);
        contentPane.add(panel);

        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBorder(new EmptyBorder(5, 5, 5, 5));
        panel_1.setBackground(new Color(240, 255, 255));
        panel_1.setBounds(0, 10, 491, 134);
        panel.add(panel_1);

        JPanel panel_2 = new JPanel();
        panel_2.setLayout(null);
        panel_2.setBackground(new Color(240, 255, 255));
        panel_2.setBounds(0, -14, 490, 85);
        panel_1.add(panel_2);

        JLabel lblInvoice = new JLabel("Invoice");
        lblInvoice.setHorizontalAlignment(SwingConstants.CENTER);
        lblInvoice.setForeground(new Color(0, 124, 124));
        lblInvoice.setFont(new Font("Bahnschrift", Font.PLAIN, 60));
        lblInvoice.setBounds(0, 0, 489, 80);
        panel_2.add(lblInvoice);

        JPanel panel_3 = new JPanel();
        panel_3.setLayout(null);
        panel_3.setBackground(Color.DARK_GRAY);
        panel_3.setBounds(0, 69, 490, 68);
        panel_1.add(panel_3);

        JButton btnPrepareAnInvoice = new JButton("Prepare an Invoice");
        btnPrepareAnInvoice.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                PrepareInvoice pi = new PrepareInvoice();
                pi.setVisible(true);
                dispose();
            }
        });
        btnPrepareAnInvoice.setForeground(Color.BLACK);
        btnPrepareAnInvoice.setFont(new Font("Bahnschrift", Font.PLAIN,
14));
        btnPrepareAnInvoice.setBackground(Color.WHITE);
        btnPrepareAnInvoice.setBounds(10, 10, 191, 35);
        panel_3.add(btnPrepareAnInvoice);

        JButton btnViewInvoices = new JButton("View Invoices");
        btnViewInvoices.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                DisplayTableInvoice d = new DisplayTableInvoice();
                d.setVisible(true);
                dispose();
            }
        });
        btnViewInvoices.setForeground(Color.BLACK);
        btnViewInvoices.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnViewInvoices.setBackground(Color.WHITE);
        btnViewInvoices.setBounds(211, 10, 158, 35);
        panel_3.add(btnViewInvoices);

        JButton button_2 = new JButton("Back");
        button_2.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                MainMenu mm = new MainMenu();
                mm.setVisible(true);
                dispose();
            }
        });
        button_2.setForeground(Color.BLACK);
        button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_2.setBackground(Color.WHITE);
        button_2.setBounds(384, 10, 97, 35);
        panel_3.add(button_2);
    }

}
```

22. Login.java

```
import java.awt.BorderLayout;
import java.awt.Color;
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 javax.swing.SwingConstants;
import java.awt.Font;
import javax.swing.JTextField;
import javax.swing.JPasswordField;
import javax.swing.JButton;
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;

public class Login extends JFrame {

    private JPanel contentPane;
    private JTextField textField;
    private JPasswordField passwordField;
    String user, pass;
    String info[][] = {{"owner", "ownerpassword"},
            {"manager", "managerpassword"}
        };
    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Login frame = new Login();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public Login() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 401, 192);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setBounds(0, 0, 387, 43);
        panel.setBackground(new Color(240, 255, 255));
        contentPane.add(panel);
        panel.setLayout(null);

        JLabel lblLogin = new JLabel("Login Page");
        lblLogin.setFont(new Font("Bahnschrift", Font.PLAIN, 30));
        lblLogin.setHorizontalAlignment(SwingConstants.CENTER);
        lblLogin.setBounds(0, 0, 387, 43);
        panel.add(lblLogin);

        JPanel panel_1 = new JPanel();
        panel_1.setBackground(Color.DARK_GRAY);
        panel_1.setBounds(0, 42, 387, 115);
        contentPane.add(panel_1);
        panel_1.setLayout(null);

        JLabel lblUsername = new JLabel("Username:");
        lblUsername.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblUsername.setForeground(Color.WHITE);
        lblUsername.setBounds(10, 10, 85, 25);
        panel_1.add(lblUsername);

        JLabel lblPassword = new JLabel("Password:");
        lblPassword.setForeground(Color.WHITE);
        lblPassword.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        lblPassword.setBounds(10, 46, 85, 25);
        panel_1.add(lblPassword);

        textField = new JTextField();
        textField.setBounds(105, 12, 272, 25);
        panel_1.add(textField);
        textField.setColumns(10);

        passwordField = new JPasswordField();
        passwordField.setBounds(105, 48, 272, 25);
        panel_1.add(passwordField);

        JButton btnLogin = new JButton("Login");
        btnLogin.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                user = textField.getText();
                pass = passwordField.~~getText()~~;
                for (int i = 0; i<2; i ++) {
                    if (user.equals(info[i][0]) &&
pass.equals(info[i][1])) {
                        if (info[i][0] == "owner") {
                            dispose();
                            GenRep mm = new GenRep();
                            mm.setVisible(true);
                        }
                        else if (info[i][0] == "manager") {
                            JOptionPane.showMessageDialog(null,
"Reports can't be accessed by managers");
                            textField.setText("");
                            passwordField.setText("");
                        }
                    }
                    else {
                        JOptionPane.showMessageDialog(null,
"Invalid username or password!");
                    }
                }
            }
        });
        btnLogin.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnLogin.setBounds(144, 83, 138, 21);
        panel_1.add(btnLogin);
    }
}
```

23. MainMenu.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
```

```java
import javax.swing.ButtonGroup;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.JToggleButton;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;

public class MainMenu extends JFrame {

	private JPanel contentPane;

	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {
				try {
					MainMenu frame = new MainMenu();
					frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}

	/**
	 * Create the frame.
	 */
	public MainMenu() {
		super("Beautymanntra || Main Menu");
		setResizable(false);
		setBounds(100, 100, 872, 290);
		contentPane = new JPanel();
		contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
		setContentPane(contentPane);
		contentPane.setLayout(null);

		JPanel panel = new JPanel();
		panel.setBackground(Color.DARK_GRAY);
		panel.setBounds(0, 126, 868, 173);
		contentPane.add(panel);
		panel.setLayout(null);

		JToggleButton tglbtnReports = new JToggleButton("Reports");
		tglbtnReports.addMouseListener(new MouseAdapter() {
			@Override
			public void mouseClicked(MouseEvent e) {
				setVisible(false);
				Reports r = new Reports();
				r.setVisible(true);
			}
			@Override
			public void mouseEntered(MouseEvent e) {
				tglbtnReports.setBackground(Color.GREEN);
			}
			@Override
```

```
        public void mouseExited(MouseEvent e) {
                tglbtnReports.setBackground(Color.WHITE);
        }
});
tglbtnReports.setForeground(Color.BLACK);
tglbtnReports.setFont(new Font("Bahnschrift", Font.PLAIN, 18));
tglbtnReports.setBackground(Color.WHITE);
tglbtnReports.setBounds(721, 0, 147, 35);
panel.add(tglbtnReports);

JToggleButton toggleButton_1 = new JToggleButton("Products");
toggleButton_1.addMouseListener(new MouseAdapter() {
        @Override
        public void mouseEntered(MouseEvent e) {
                toggleButton_1.setBackground(Color.GREEN);
        }
        @Override
        public void mouseExited(MouseEvent e) {
                toggleButton_1.setBackground(Color.WHITE);
        }
        @Override
        public void mouseClicked(MouseEvent e) {
                Products p = new Products();
                setVisible(false);
                p.setVisible(true);
        }
});

JToggleButton toggleButton_4 = new JToggleButton("Clients");
toggleButton_4.addMouseListener(new MouseAdapter() {
        @Override
        public void mouseEntered(MouseEvent e) {
                toggleButton_4.setBackground(Color.GREEN);
        }
        @Override
        public void mouseExited(MouseEvent e) {
                toggleButton_4.setBackground(Color.WHITE);
        }
        @Override
        public void mouseClicked(MouseEvent e) {
                Clients c = new Clients();
                setVisible(false);
                c.setVisible(true);
        }
});
toggleButton_4.setForeground(Color.BLACK);
toggleButton_4.setFont(new Font("Bahnschrift", Font.PLAIN, 18));
toggleButton_4.setBackground(Color.WHITE);
toggleButton_4.setBounds(0, 0, 147, 35);
panel.add(toggleButton_4);

toggleButton_1.setForeground(Color.BLACK);
toggleButton_1.setFont(new Font("Bahnschrift", Font.PLAIN, 18));
toggleButton_1.setBackground(Color.WHITE);
toggleButton_1.setBounds(145, 0, 147, 35);
panel.add(toggleButton_1);

JToggleButton toggleButton_2 = new JToggleButton("Services");
toggleButton_2.addMouseListener(new MouseAdapter() {
     @Override
     public void mouseEntered(MouseEvent e) {
             toggleButton_2.setBackground(Color.GREEN);
     }
     @Override
     public void mouseExited(MouseEvent e) {
             toggleButton_2.setBackground(Color.WHITE);
     }
     @Override
     public void mouseClicked(MouseEvent e) {
             Services s = new Services();
             setVisible(false);
             s.setVisible(true);
     }
});
toggleButton_2.setForeground(Color.BLACK);
toggleButton_2.setFont(new Font("Bahnschrift", Font.PLAIN, 18));
toggleButton_2.setBackground(Color.WHITE);
toggleButton_2.setBounds(290, 0, 147, 35);
panel.add(toggleButton_2);

JToggleButton tglbtnAppointments = new JToggleButton("Invoice");
tglbtnAppointments.addMouseListener(new MouseAdapter() {
     @Override
     public void mouseEntered(MouseEvent e) {
             tglbtnAppointments.setBackground(Color.GREEN);
     }
     @Override
     public void mouseExited(MouseEvent e) {
             tglbtnAppointments.setBackground(Color.WHITE);
     }
     @Override
     public void mouseClicked(MouseEvent e) {
             Invoices i = new Invoices();
             i.setVisible(true);
             dispose();
     }
});
tglbtnAppointments.setForeground(Color.BLACK);
tglbtnAppointments.setFont(new Font("Bahnschrift", Font.PLAIN, 18));
tglbtnAppointments.setBackground(Color.WHITE);
tglbtnAppointments.setBounds(435, 0, 147, 35);
panel.add(tglbtnAppointments);

JToggleButton toggleButton_5 = new JToggleButton("Branches");
toggleButton_5.addMouseListener(new MouseAdapter() {
     @Override
     public void mouseClicked(MouseEvent e) {
             Branches b = new Branches();
             setVisible(false);
             b.setVisible(true);
     }
     @Override
     public void mouseEntered(MouseEvent e) {
             toggleButton_5.setBackground(Color.GREEN);
     }
     @Override
     public void mouseExited(MouseEvent e) {
             toggleButton_5.setBackground(Color.WHITE);
```

```
    }
});
toggleButton_5.setForeground(Color.BLACK);
toggleButton_5.setFont(new Font("Bahnschrift", Font.PLAIN, 18));
toggleButton_5.setBackground(Color.WHITE);
toggleButton_5.setBounds(580, 0, 147, 35);
panel.add(toggleButton_5);

JPanel panel_1 = new JPanel();
panel_1.setLayout(null);
panel_1.setBackground(new Color(240, 255, 255));
panel_1.setBounds(0, 0, 868, 127);
contentPane.add(panel_1);

JLabel label = new JLabel("BEAUTY\r\n");
label.setHorizontalAlignment(SwingConstants.CENTER);
label.setForeground(Color.CYAN.darker().darker());
label.setFont(new Font("Segoe UI Symbol", Font.BOLD, 70));
label.setBounds(0, -11, 868, 66);
panel_1.add(label);

JLabel label_1 = new JLabel("MANNTRA\r\n");
label_1.setHorizontalAlignment(SwingConstants.CENTER);
label_1.setForeground(Color.CYAN.darker().darker());
label_1.setFont(new Font("Segoe UI Symbol", Font.PLAIN, 55));
label_1.setBounds(0, 40, 868, 60);
panel_1.add(label_1);

JLabel label_2 = new JLabel("SALON\r\n");
label_2.setHorizontalAlignment(SwingConstants.CENTER);
label_2.setForeground(Color.CYAN.darker());
label_2.setFont(new Font("Segoe UI Symbol", Font.PLAIN, 40));
label_2.setBounds(0, 79, 868, 54);
panel_1.add(label_2);

ButtonGroup group = new ButtonGroup();
group.add(toggleButton_5);
group.add(tglbtnAppointments);
group.add(toggleButton_2);
group.add(toggleButton_1);
group.add(tglbtnReports);

JToggleButton toggleButton_3 = new JToggleButton("Branches");
toggleButton_3.setForeground(Color.BLACK);
toggleButton_3.setFont(new Font("Bahnschrift", Font.PLAIN, 18));
toggleButton_3.setBackground(Color.WHITE);
toggleButton_3.setBounds(0, 0, 147, 35);
panel.add(toggleButton_3);

JPanel panel_2 = new RoundedPanel(200, Color.GREEN.darker());
panel_2.setBackground(Color.DARK_GRAY);
panel_2.setBounds(10, 45, 88, 79);
panel.add(panel_2);
panel_2.setLayout(null);

JLabel lblNewLabel_1 = new JLabel("MON");
lblNewLabel_1.setForeground(Color.WHITE);
lblNewLabel_1.setHorizontalAlignment(SwingConstants.CENTER);
lblNewLabel_1.setFont(new Font("Bahnschrift", Font.BOLD, 30));
```

```java
lblNewLabel_1.setBounds(10, 0, 68, 59);
panel_2.add(lblNewLabel_1);

JLabel lblNewLabel_2 = new JLabel("10:00 to 20:30");
lblNewLabel_2.setForeground(Color.WHITE);
lblNewLabel_2.setFont(new Font("Bahnschrift", Font.BOLD, 10));
lblNewLabel_2.setHorizontalAlignment(SwingConstants.CENTER);
lblNewLabel_2.setBounds(10, 42, 68, 27);
panel_2.add(lblNewLabel_2);

JPanel panel_3 = new RoundedPanel(200, Color.GREEN.darker());
panel_3.setBackground(Color.DARK_GRAY);
panel_3.setBounds(132, 45, 88, 79);
panel.add(panel_3);
panel_3.setLayout(null);

JLabel lblTue = new JLabel("TUE");
lblTue.setForeground(Color.WHITE);
lblTue.setHorizontalAlignment(SwingConstants.CENTER);
lblTue.setFont(new Font("Bahnschrift", Font.BOLD, 30));
lblTue.setBounds(10, 0, 68, 59);
panel_3.add(lblTue);

JLabel label_3 = new JLabel("10:00 to 20:30");
label_3.setHorizontalAlignment(SwingConstants.CENTER);
label_3.setForeground(Color.WHITE);
label_3.setFont(new Font("Bahnschrift", Font.BOLD, 10));
label_3.setBounds(10, 42, 68, 27);
panel_3.add(label_3);

JPanel panel_4 = new RoundedPanel(200, Color.GREEN.darker());
panel_4.setBackground(Color.DARK_GRAY);
panel_4.setBounds(251, 45, 88, 79);
panel.add(panel_4);
panel_4.setLayout(null);

JLabel lblWed = new JLabel("WED");
lblWed.setForeground(Color.WHITE);
lblWed.setHorizontalAlignment(SwingConstants.CENTER);
lblWed.setFont(new Font("Bahnschrift", Font.BOLD, 30));
lblWed.setBounds(10, 0, 68, 59);
panel_4.add(lblWed);

JLabel label_4 = new JLabel("10:00 to 20:30");
label_4.setHorizontalAlignment(SwingConstants.CENTER);
label_4.setForeground(Color.WHITE);
label_4.setFont(new Font("Bahnschrift", Font.BOLD, 10));
label_4.setBounds(10, 42, 68, 27);
panel_4.add(label_4);

JPanel panel_5 = new RoundedPanel(200, Color.GREEN.darker());
panel_5.setBackground(Color.DARK_GRAY);
panel_5.setBounds(381, 45, 88, 79);
panel.add(panel_5);
panel_5.setLayout(null);

JLabel lblThur = new JLabel("THU");
lblThur.setForeground(Color.WHITE);
lblThur.setHorizontalAlignment(SwingConstants.CENTER);
lblThur.setFont(new Font("Bahnschrift", Font.BOLD, 30));
lblThur.setBounds(10, 0, 68, 59);
panel_5.add(lblThur);

JLabel label_5 = new JLabel("10:00 to 20:30");
label_5.setHorizontalAlignment(SwingConstants.CENTER);
label_5.setForeground(Color.WHITE);
label_5.setFont(new Font("Bahnschrift", Font.BOLD, 10));
label_5.setBounds(10, 42, 68, 27);
panel_5.add(label_5);

JPanel panel_6 = new RoundedPanel(200, Color.GREEN.darker());
panel_6.setBackground(Color.DARK_GRAY);
panel_6.setBounds(494, 45, 88, 79);
panel.add(panel_6);
panel_6.setLayout(null);

JLabel lblFri = new JLabel("FRI");
lblFri.setForeground(Color.WHITE);
lblFri.setHorizontalAlignment(SwingConstants.CENTER);
lblFri.setFont(new Font("Bahnschrift", Font.BOLD, 30));
lblFri.setBounds(10, 0, 68, 59);
panel_6.add(lblFri);

JLabel label_6 = new JLabel("10:00 to 20:30");
label_6.setHorizontalAlignment(SwingConstants.CENTER);
label_6.setForeground(Color.WHITE);
label_6.setFont(new Font("Bahnschrift", Font.BOLD, 10));
label_6.setBounds(10, 42, 68, 27);
panel_6.add(label_6);

JPanel panel_7 = new RoundedPanel(200, Color.GREEN.darker());
panel_7.setBackground(Color.DARK_GRAY);
panel_7.setBounds(625, 45, 88, 79);
panel.add(panel_7);
panel_7.setLayout(null);

JLabel lblSat = new JLabel("SAT");
lblSat.setForeground(Color.WHITE);
lblSat.setHorizontalAlignment(SwingConstants.CENTER);
lblSat.setFont(new Font("Bahnschrift", Font.BOLD, 30));
lblSat.setBounds(10, 0, 68, 59);
panel_7.add(lblSat);

JLabel label_7 = new JLabel("10:00 to 20:30");
label_7.setHorizontalAlignment(SwingConstants.CENTER);
label_7.setForeground(Color.WHITE);
label_7.setFont(new Font("Bahnschrift", Font.BOLD, 10));
label_7.setBounds(10, 42, 68, 27);
panel_7.add(label_7);

JPanel panel_8 = new RoundedPanel(200, Color.GREEN.darker());
panel_8.setBackground(Color.DARK_GRAY);
panel_8.setBounds(753, 45, 88, 79);
panel.add(panel_8);
panel_8.setLayout(null);

JLabel lblSun = new JLabel("SUN");
lblSun.setForeground(Color.WHITE);
```

```
        lblSun.setHorizontalAlignment(SwingConstants.CENTER);
        lblSun.setFont(new Font("Bahnschrift", Font.BOLD, 30));
        lblSun.setBounds(10, 0, 68, 59);
        panel_8.add(lblSun);

        JLabel label_8 = new JLabel("10:00 to 20:30");
        label_8.setHorizontalAlignment(SwingConstants.CENTER);
        label_8.setForeground(Color.WHITE);
        label_8.setFont(new Font("Bahnschrift", Font.BOLD, 10));
        label_8.setBounds(10, 42, 68, 27);
        panel_8.add(label_8);
    }
}
```

24. PrepareInvoice.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.Insets;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.io.FileOutputStream;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Time;
import java.text.SimpleDateFormat;
import java.time.LocalDateTime;
import java.time.format.DateTimeFormatter;
import java.util.ArrayList;
import java.util.Calendar;
import java.util.Date;

import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JFileChooser;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JScrollPane;
import javax.swing.JTable;
import javax.swing.JTextField;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;
import javax.swing.table.DefaultTableModel;
import com.itextpdf.text.BaseColor;
import com.itextpdf.text.Document;
import com.itextpdf.text.Element;
import com.itextpdf.text.FontFactory;
import com.itextpdf.text.Paragraph;
import com.itextpdf.text.Phrase;
import com.itextpdf.text.pdf.BaseFont;
import com.itextpdf.text.pdf.PdfPCell;
```

```java
import com.itextpdf.text.pdf.PdfPTable;
import com.itextpdf.text.pdf.PdfWriter;
import javax.swing.JRadioButton;

public class PrepareInvoice extends JFrame {

    private JPanel contentPane;
    private JTextField clientNumber;
    JLabel cname, cnumber;
    private JTable table;
    JComboBox serviceComboBox, prodComboBox, comboBox_2;
    int selectedRowIndex;
    JButton prodButton, servButton;
    String path1;
    JFileChooser j;
    ArrayList<RowDataService> servlist;
    ArrayList<RowDataProduct> prodlist;
    JLabel lblTotalAmount;
    JButton btnDeleteRow, btnNewButton_1;
    String query, query1, i, ii;
    Connection con, con1, con11;
    ResultSet rs, rs1, prodrs, servrs;
    PreparedStatement pst, pst1;
    int noOfProds = 0, noOfServs = 0, time, g;
    String path;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    PrepareInvoice frame = new PrepareInvoice();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */

    public void setComboButtonEditable(boolean n) {
        serviceComboBox.setEnabled(n);
        prodComboBox.setEnabled(n);
        prodButton.setEnabled(n);
        servButton.setEnabled(n);
        comboBox_2.setEnabled(n);
        btnNewButton_1.setEnabled(n);
    }

    @SuppressWarnings("serial")
    public PrepareInvoice() {
        setResizable(false);
        setBounds(100, 100, 1018, 548);
        contentPane = new JPanel();
        contentPane.setBackground(Color.CYAN.darker().darker().darker());
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JLabel lblPrepareAnInvoice = new JLabel("Prepare an Invoice");
        lblPrepareAnInvoice.setHorizontalAlignment(SwingConstants.CENTER);
        lblPrepareAnInvoice.setForeground(Color.WHITE);
        lblPrepareAnInvoice.setFont(new Font("Bahnschrift", Font.PLAIN,
26));
        lblPrepareAnInvoice.setBounds(0, 0, 1004, 30);
        contentPane.add(lblPrepareAnInvoice);

        JLabel lblServiceTaken = new JLabel("Service Taken:");
        lblServiceTaken.setHorizontalAlignment(SwingConstants.RIGHT);
        lblServiceTaken.setForeground(Color.WHITE);
        lblServiceTaken.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        lblServiceTaken.setBounds(10, 136, 103, 21);
        contentPane.add(lblServiceTaken);

        serviceComboBox = new JComboBox();
        serviceComboBox.setBackground(Color.WHITE);
        serviceComboBox.setFont(new Font("Arial", Font.PLAIN, 12));
        serviceComboBox.addItem("Select a service");
        try {
            query = "SELECT Subcategory, SerialNo FROM servicelist";
            con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
            pst = con.prepareStatement(query);
            rs = pst.executeQuery();
            while(rs.next()) {
                serviceComboBox.addItem(rs.getString("Subcategory"));
            }
        }
        catch(Exception e){
            JOptionPane.showMessageDialog(null, e);
        }
        serviceComboBox.setBounds(123, 138, 255, 21);
        contentPane.add(serviceComboBox);

        JLabel lblProductBought = new JLabel("Product bought:");
        lblProductBought.setHorizontalAlignment(SwingConstants.RIGHT);
        lblProductBought.setForeground(Color.WHITE);
        lblProductBought.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        lblProductBought.setBounds(10, 105, 103, 21);
        contentPane.add(lblProductBought);

        prodComboBox = new JComboBox();
        prodComboBox.setBackground(Color.WHITE);
        prodComboBox.setFont(new Font("Arial", Font.PLAIN, 12));
        prodComboBox.addItem("Select a Product");
        try {
            query = "SELECT Company, ProductName FROM productlist";
            con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
            pst = con.prepareStatement(query);
            rs = pst.executeQuery();
            while(rs.next()) {
                prodComboBox.addItem(rs.getString("ProductName"));
            }
        }
        catch(Exception e1) {
            JOptionPane.showMessageDialog(null, e1);
        }
        prodComboBox.setBounds(123, 107, 255, 21);
        contentPane.add(prodComboBox);

        JLabel lblClientId = new JLabel("Client Mobile:");
        lblClientId.setHorizontalAlignment(SwingConstants.RIGHT);
        lblClientId.setForeground(Color.WHITE);
        lblClientId.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        lblClientId.setBounds(10, 72, 103, 21);
        contentPane.add(lblClientId);

        clientNumber = new JTextField();
        clientNumber.setBounds(123, 74, 255, 21);
        contentPane.add(clientNumber);
        clientNumber.setColumns(10);

        JButton clientButton = new JButton("+");
        clientButton.setBackground(Color.WHITE);
        clientButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                if (clientNumber.getText().isEmpty()==false) {
                    query1 = "SELECT Name, ClientID FROM 
csiatables.clientlist WHERE MobileNumber=?";
                    try {
                        con1 = 
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root", 
"mySQLpassword");
                        pst1 = con1.prepareStatement(query1);
                        pst1.setString(1, clientNumber.getText());
                        rs1 = pst1.executeQuery();
                        if(rs1.next()) {

    cname.setText(rs1.getString("Name"));
    cnumber.setText(rs1.getString("ClientID"));
                            clientNumber.setEditable(false);
                            setComboButtonEditable(true);
                        }
                        else {
                            JOptionPane.showMessageDialog(null, 
"No client registered with that phone number! Add Client");
                            AddClient client = new AddClient();
                            client.setVisible(true);
                        }
                    } catch (SQLException e1) {
                            e1.printStackTrace();
                    }
                }
                else {
                    JOptionPane.showMessageDialog(null, "Enter Client 
Phone number");
                }
            }
        });
        clientButton.setMargin(new Insets(0, 0, 0, 0));
        clientButton.setFont(new Font("Tahoma", Font.BOLD, 18));
        clientButton.setBounds(388, 74, 21, 21);
        contentPane.add(clientButton);

        JLabel lblClientName = new JLabel("Client Name:");
        lblClientName.setHorizontalAlignment(SwingConstants.RIGHT);
        lblClientName.setForeground(Color.WHITE);
        lblClientName.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        lblClientName.setBounds(713, 105, 101, 21);
        contentPane.add(lblClientName);

        prodButton = new JButton("+");
        prodButton.setBackground(Color.WHITE);
        prodButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                ii = prodComboBox.getSelectedItem().toString();
                if (ii != "Select a Product") {
                    noOfProds++;
                    con11 = null;
                    try {
                        prodlist = new
ArrayList<RowDataProduct>();
                        con11 =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                        query1 = "select Company, ProductName,
Size, Rate from productlist where ProductName = '" + ii + "'";
                        pst = con11.prepareStatement(query1);
                        prodrs = pst.executeQuery();
                        if (prodrs.next()) {
                            int qty =
Integer.parseInt(comboBox_2.getSelectedItem().toString());
                            RowDataProduct product = new
RowDataProduct("Product", ii, qty, prodrs.getDouble("Rate"),
prodrs.getDouble("Rate") * qty);
                            prodlist.add(product);
                            addProductRowToTable(prodlist);
                            getTotalAmount();
                            updateLabel();
                        }
                    }
                    catch(Exception e2) {
                        JOptionPane.showMessageDialog(null, e2);
                    }
                }
                else {
                    JOptionPane.showMessageDialog(null, "Select
products bought by customer");
                }
            }
        });
        prodButton.setMargin(new Insets(0, 0, 0, 0));
        prodButton.setFont(new Font("Tahoma", Font.BOLD, 18));
        prodButton.setBounds(475, 107, 21, 21);
        contentPane.add(prodButton);

        servButton = new JButton("+");
        servButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                i = serviceComboBox.getSelectedItem().toString();
                if (i != "Select a service") {
                    noOfServs++;
                    con1 = null;
                    try {
                        servlist = new
ArrayList<RowDataService>();
                        con1 =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                        query1 = "select Category, Rate from
servicelist where Subcategory = '" + i + "'";
                        pst = con1.prepareStatement(query1);
                        servrs = pst.executeQuery();
                        if (servrs.next()) {
                            RowDataService service = new
RowDataService("Service", i, 1, servrs.getDouble("Rate"),
servrs.getDouble("Rate"));
                            servlist.add(service);
                            addServiceRowToTable(servlist);
                            getTotalAmount();
                            updateLabel();
                        }
                    }
                    catch(Exception e2) {
                        JOptionPane.showMessageDialog(null, e2);
                    }
                }
                else {
                    JOptionPane.showMessageDialog(null, "Select
services");
                }
            }
        });
        servButton.setBackground(Color.WHITE);
        servButton.setMargin(new Insets(0, 0, 0, 0));
        servButton.setFont(new Font("Tahoma", Font.BOLD, 18));
        servButton.setBounds(388, 138, 21, 21);
        contentPane.add(servButton);

        comboBox_2 = new JComboBox();
        for (int j = 1; j < 100; j ++) {
            comboBox_2.addItem(j);
        }
        comboBox_2.setFont(new Font("Arial", Font.PLAIN, 12));
        comboBox_2.setBackground(Color.WHITE);
        comboBox_2.setBounds(431, 107, 34, 21);
        contentPane.add(comboBox_2);

        Date myDateObj = new Date();
        SimpleDateFormat formatter = new SimpleDateFormat("dd/MM/yyyy
HH:mm:ss");
       java.sql.Date formattedDate = new java.sql.Date(myDateObj.getTime());

        JLabel lblDate = new JLabel("Invoice Date: " +
formatter.format(myDateObj));
        lblDate.setHorizontalAlignment(SwingConstants.RIGHT);
```

```
             lblDate.setForeground(Color.WHITE);
             lblDate.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
             lblDate.setBounds(733, 40, 248, 21);
             contentPane.add(lblDate);

             cname = new JLabel("");
             cname.setHorizontalAlignment(SwingConstants.LEFT);
             cname.setForeground(Color.WHITE);
             cname.setFont(new Font("Bahnschrift", Font.BOLD, 14));
             cname.setBounds(824, 105, 165, 21);
             contentPane.add(cname);

             JLabel lblClientPhoneNumber = new JLabel("Client ID:");
             lblClientPhoneNumber.setHorizontalAlignment(SwingConstants.RIGHT);
             lblClientPhoneNumber.setForeground(Color.WHITE);
             lblClientPhoneNumber.setFont(new Font("Bahnschrift", Font.PLAIN,
14));
             lblClientPhoneNumber.setBounds(713, 72, 101, 21);
             contentPane.add(lblClientPhoneNumber);

             cnumber = new JLabel("");
             cnumber.setHorizontalAlignment(SwingConstants.LEFT);
             cnumber.setForeground(Color.WHITE);
             cnumber.setFont(new Font("Bahnschrift", Font.BOLD, 14));
             cnumber.setBounds(824, 74, 170, 21);
             contentPane.add(cnumber);

             JScrollPane scrollPane = new JScrollPane();
             scrollPane.setBounds(10, 169, 984, 283);
             contentPane.add(scrollPane);

             table = new JTable();
             DefaultTableModel model = new DefaultTableModel();
             model.setRowCount(10);
             table.addMouseListener(new MouseAdapter() {
                    @Override
                    public void mouseClicked(MouseEvent e) {
                           btnDeleteRow.setVisible(true);
                    }
             });
             table.setDefaultEditor(Object.class, null);
             table.setShowGrid(false);
             table.setModel(new DefaultTableModel(
                    new Object[][] {
                    },
                    new String[] {
                           "Item Type", "Description", "Quantity", "Rate(in
\u20B9)", "Amount(in \u20B9)"
                    }
             ) {
                    Class[] columnTypes = new Class[] {
                           String.class, String.class, Integer.class,
Double.class, Double.class
                    };
                    public Class getColumnClass(int columnIndex) {
                           return columnTypes[columnIndex];
                    }
             });
             scrollPane.setViewportView(table);
```

```
        btnDeleteRow = new JButton("Delete row");
        btnDeleteRow.setVisible(false);
        btnDeleteRow.addActionListener(new ActionListener() {
            DefaultTableModel model = (DefaultTableModel)
table.getModel();
            public void actionPerformed(ActionEvent e) {
                try {
                    selectedRowIndex = table.getSelectedRow();
                    if (table.getValueAt(selectedRowIndex,
0).equals("Product")) {
                        noOfProds--;
                    }
                    else if (table.getValueAt(selectedRowIndex,
0).equals("Service")) {
                        noOfServs--;
                    }
                    subtractValue(selectedRowIndex);
                    model.removeRow(selectedRowIndex);
                }
                catch(Exception ex) {
                    JOptionPane.showMessageDialog(null, "Error");
                }
            }
        });
        btnDeleteRow.setMargin(new Insets(0, 0, 0, 0));
        btnDeleteRow.setFont(new Font("Bahnschrift", Font.PLAIN, 18));
        btnDeleteRow.setBackground(Color.WHITE);
        btnDeleteRow.setBounds(419, 138, 136, 21);
        contentPane.add(btnDeleteRow);

        JComboBox branchInfo = new JComboBox();
        try {
            query = "SELECT Address, ContactNumber FROM branchlist";
            con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
            pst = con.prepareStatement(query);
            rs = pst.executeQuery();
            while(rs.next()) {
                branchInfo.addItem(rs.getString("Address") + " - " +
rs.getString("ContactNumber"));
            }
        }
        catch(Exception e1) {
            JOptionPane.showMessageDialog(null, e1);
        }
        branchInfo.setFont(new Font("Arial", Font.PLAIN, 12));
        branchInfo.setBackground(Color.WHITE);
        branchInfo.setBounds(123, 43, 575, 21);
        contentPane.add(branchInfo);

        JButton brButton = new JButton("+");
        brButton.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                JOptionPane.showMessageDialog(null, "Selected Branch
Successfully");
            }
        });
```

```
             brButton.setMargin(new Insets(0, 0, 0, 0));
             brButton.setFont(new Font("Tahoma", Font.BOLD, 18));
             brButton.setBackground(Color.WHITE);
             brButton.setBounds(708, 43, 21, 21);
             contentPane.add(brButton);

             btnNewButton_1 = new JButton("Generate an Invoice");
             btnNewButton_1.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                          // Generating pdf
                          path1="";
                          j = new JFileChooser();
                          j.setFileSelectionMode(JFileChooser.DIRECTORIES_ONLY);
                          g = j.showSaveDialog(null);

                          if(g==JFileChooser.APPROVE_OPTION) {
                                path1 = j.getSelectedFile().toString();
                          }
                          time = (int) (System.currentTimeMillis()/1000);
                          path1 = path1.replace("\\", "\\\\");
                          path = path1+"\\\\Invoice-"+cname.getText()+"-
"+time+".pdf";

                          //Insert invoice info to database

                          Document doc = new Document();
                          try{
                                if (table.getRowCount()!=0) {
                                       String FONT1 =
"resources/fonts/PlayfairDisplay-Regular.ttf";
                                       com.itextpdf.text.Font f3
=FontFactory.getFont(FONT1, BaseFont.IDENTITY_H, BaseFont.EMBEDDED, 18);
                                       PdfWriter.getInstance(doc, new
FileOutputStream(path));

                                       doc.open();
                                       Paragraph para = new
Paragraph("BEAUTYMANNTRA INVOICE",

      FontFactory.getFont(FontFactory.HELVETICA_BOLD, 24,
                                                                 Font.BOLD,
BaseColor.CYAN.darker().darker().darker()));
                                       para.setAlignment(Element.ALIGN_CENTER);
                                       Paragraph para1 = new Paragraph("---------
-------------------------------------------------"
                                                    + "-------------------------
----------------------------------------------");
                                       Paragraph para2 = new Paragraph("Date: " +
formatter.format(myDateObj));
                                       para2.setAlignment(Element.ALIGN_RIGHT);
                                       Paragraph para3 = new Paragraph("---------
-------------------------------------------------"
                                                    + "-------------------------
----------------------------------------------");
                                       Paragraph para4 = new Paragraph("Client
ID: " + cnumber.getText());
                                       Paragraph para5 = new Paragraph("Client
Name: " + cname.getText());
```

```
                                       Paragraph para6 = new Paragraph("---------
------------------------------------------------"
                                                     + "--------------------------
----------------------------------------------");
                                       doc.add(para); doc.add(para1);
doc.add(para2);
                                       doc.add(para3); doc.add(para4);
doc.add(para5); doc.add(para6);

                                       PdfPTable table1 = new PdfPTable(5);
                                       table1.setWidthPercentage(105);
                                       table1.setSpacingBefore(11f);
                                       table1.setSpacingAfter(11f);

                                       float[] colWidth= {2f, 2f, 2f, 2f, 2f};
                                       table1.setWidths(colWidth);
                                       PdfPCell c1 = new PdfPCell(new
Phrase("Item Type"));
                                       table1.addCell(c1);
                                       PdfPCell c2 = new PdfPCell(new
Phrase("Description"));
                                       table1.addCell(c2);
                                       PdfPCell c3 = new PdfPCell(new
Phrase("Quantity"));
                                       table1.addCell(c3);
                                       PdfPCell c4 = new PdfPCell(new
Phrase("Rate"));
                                       table1.addCell(c4);
                                       PdfPCell c5 = new PdfPCell(new
Phrase("Amount"));
                                       table1.addCell(c5);
                                       for (int m = 0; m < table.getRowCount(); m
++) {
                                              String itype = table.getValueAt(m,
0).toString();
                                              String desc = table.getValueAt(m,
1).toString();
                                              String qty = table.getValueAt(m,
2).toString();
                                              String r = table.getValueAt(m,
3).toString();
                                              String amt = table.getValueAt(m,
4).toString();
                                              table1.addCell(itype);
table1.addCell(desc); table1.addCell(qty);
                                              table1.addCell(r);
table1.addCell(amt);
                                       }
                                       doc.add(table1);
                                       Paragraph para7 = new Paragraph("Total
Amount: Rs. " + sum, f3);
                                       para7.setAlignment(Element.ALIGN_RIGHT);
                                       doc.add(para7);

                                       Paragraph space = new Paragraph("---------
------------------------------------------------"
                                                     + "--------------------------
----------------------------------------------");
```

```
                                        Paragraph branchinfo = new
Paragraph(branchInfo.getSelectedItem().toString(),

       FontFactory.getFont(FontFactory.HELVETICA, 10, Font.BOLD,
BaseColor.BLACK));

       branchinfo.setAlignment(Element.ALIGN_CENTER);
                                        doc.add(space);
                                        doc.add(branchinfo);
                                        doc.close();

                                        JOptionPane.showMessageDialog(null, "PDF
Generated for Invoice");
                                        try {
                                               String invoiceinfo = "INSERT INTO
`csiatables`.`invoicelist` (`Path`, `ClientName`, `Date`, `Expense`) VALUES "
                                                            + "('"+path+"',
'"+cname.getText()+"', '"+formattedDate+"', '"+sum+"')";
                                               Connection con =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                                               PreparedStatement invoicepst =
con.prepareStatement(invoiceinfo);
                                               invoicepst.executeUpdate();
                                        }
                                        catch(Exception e11) {
                                               JOptionPane.showMessageDialog(null,
e11);
                                        }

                                        //Insert product and service data in
database table

                                        try {
                                               String inv = "select SerialNo from
invoicelist where Date = '"+formattedDate+"'";

                                               String prodInfo = "INSERT INTO
`csiatables`.`invoiceproductlist` (`invoiceno`, `Date`, `Product Name`,
`productID`, `price`) VALUES (?, ?, ?, ?, ?)";
                                               String servInfo = "INSERT INTO
`csiatables`.`invoiceservicelist` (`invoiceno`, `Date`, `Service Name`,
`serviceID`, `price`) VALUES (?, ?, ?, ?, ?)";
                                               Connection con1 =
DriverManager.getConnection("jdbc:mysql://localhost:3306/csiatables", "root",
"mySQLpassword");
                                               for (int i = 0; i <
table.getRowCount(); i++) {
                                                      PreparedStatement prodpst =
con1.prepareStatement(prodInfo);
                                                      PreparedStatement invIdpst =
con1.prepareStatement(inv);
                                                      ResultSet rs1 =
invIdpst.executeQuery();
                                                      int id = 0;
                                                      while (rs1.next()) {
                                                             id =
rs1.getInt("SerialNo");
                                                      }
```

```
                          if (table.getValueAt(i,
0).toString().equals("Product")) {
                             String prod = "select
SerialNo from productlist where ProductName = '"+table.getValueAt(i,
1).toString()+"'";

                             for (int Quantity = 0;
Quantity < Integer.parseInt(table.getValueAt(i, 2).toString()); Quantity++) {

   PreparedStatement prodIdpst = con1.prepareStatement(prod);
                                 ResultSet rsprod
= prodIdpst.executeQuery();
                                 int prodId = 0;
                                 while
(rsprod.next()) {
                                    prodId =
rsprod.getInt("SerialNo");
                                 }
                                 double price =
Double.parseDouble(table.getValueAt(i, 3).toString());
                                 String prodName
= table.getValueAt(i, 1).toString();

   prodpst.setInt(1, id);

   prodpst.setDate(2, formattedDate);

   prodpst.setString(3, prodName);

   prodpst.setInt(4, prodId);

   prodpst.setDouble(5, price);

   prodpst.executeUpdate();
                             }
                          }
                          else if (table.getValueAt(i,
0).toString().equals("Service")) {
                             String serv = "select
SerialNo from servicelist where Subcategory = '"+table.getValueAt(i,
1).toString()+"'";
                             PreparedStatement
servpst = con1.prepareStatement(servInfo);
                             PreparedStatement
servIdpst = con1.prepareStatement(serv);
                             ResultSet rsserv =
servIdpst.executeQuery();
                             int servId = 0;
                             while (rsserv.next()) {
                                servId =
rsserv.getInt("SerialNo");
                             }
                             double amount =
Double.parseDouble(table.getValueAt(i, 4).toString());
                             String servName =
table.getValueAt(i, 1).toString();
                             servpst.setInt(1, id);
                             servpst.setDate(2,
formattedDate);
```

```
                                    servpst.setString(3,
servName);
                                    servpst.setInt(4,
servId);
                                    servpst.setDouble(5,
amount);
    servpst.executeUpdate();
                                }
                            }
                        }
                        catch(Exception e1){
                            JOptionPane.showMessageDialog(null,
e1);
                        }

                    }
                    else{
                        JOptionPane.showMessageDialog(null, "Enter
services taken or products bought");
                    }
                }
                catch(Exception e1) {
                    JOptionPane.showMessageDialog(null, e1);
                }
            }
        });
        btnNewButton_1.setBackground(Color.WHITE);
        btnNewButton_1.setFont(new Font("Bahnschrift", Font.PLAIN, 20));
        btnNewButton_1.setBounds(10, 462, 243, 30);
        contentPane.add(btnNewButton_1);

        setComboButtonEditable(false);

        JLabel lblQty = new JLabel("Qty:");
        lblQty.setHorizontalAlignment(SwingConstants.RIGHT);
        lblQty.setForeground(Color.WHITE);
        lblQty.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        lblQty.setBounds(387, 105, 34, 21);
        contentPane.add(lblQty);

        lblTotalAmount = new JLabel("Total Amount to be paid: ");
        lblTotalAmount.setHorizontalAlignment(SwingConstants.LEFT);
        lblTotalAmount.setForeground(Color.WHITE);
        lblTotalAmount.setFont(new Font("Bahnschrift", Font.PLAIN, 26));
        lblTotalAmount.setBounds(475, 449, 529, 30);
        contentPane.add(lblTotalAmount);

        JButton btnBack = new JButton("Back");
        btnBack.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                dispose();
                Reports rp = new Reports();
                rp.setVisible(true);
            }
        });
        btnBack.setFont(new Font("Bahnschrift", Font.PLAIN, 20));
        btnBack.setBackground(Color.WHITE);
        btnBack.setBounds(263, 462, 110, 30);
        contentPane.add(btnBack);

        JLabel lblBranchAddress = new JLabel("Branch Info: ");
        lblBranchAddress.setHorizontalAlignment(SwingConstants.RIGHT);
        lblBranchAddress.setForeground(Color.WHITE);
        lblBranchAddress.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        lblBranchAddress.setBounds(0, 41, 126, 21);
        contentPane.add(lblBranchAddress);
    }

    public class RowDataService{
        String itemtype, description;
        double rate, amount;
        int quantity;

        public RowDataService(String ItemType, String Description, int
Quantity, double Rate, double Amount) {
            this.itemtype=ItemType;
            this.description=Description;
            this.quantity=Quantity;
            this.rate=Rate;
            this.amount = Amount;
        }
    }

    public class RowDataProduct{
        String itemtype, description;
        double rate, amount;
        int quantity;

        public RowDataProduct(String ItemType, String Description, int
Quantity, double Rate, double Amount) {
            this.itemtype=ItemType;
            this.description=Description;
            this.quantity=Quantity;
            this.rate=Rate;
            this.amount = Amount;
        }
    }

    public void addServiceRowToTable(ArrayList<RowDataService> list) {
        DefaultTableModel dtm = (DefaultTableModel) table.getModel();
        ArrayList<RowDataService> sl = list;
        Object rowData[] = new Object[5];
        for (int i = 0; i < sl.size(); i ++) {
            rowData[0] = sl.get(i).itemtype;
            rowData[1] = sl.get(i).description;
            rowData[2] = sl.get(i).quantity;
            rowData[3] = sl.get(i).rate;
            rowData[4] = sl.get(i).amount;
            dtm.addRow(rowData);
        }
    }

    public void addProductRowToTable(ArrayList<RowDataProduct> list) {
        DefaultTableModel dtm = (DefaultTableModel) table.getModel();
        ArrayList<RowDataProduct> sl = list;
        Object rowData[] = new Object[5];
        for (int i = 0; i < sl.size(); i ++) {
            rowData[0] = sl.get(i).itemtype;
            rowData[1] = sl.get(i).description;
            rowData[2] = sl.get(i).quantity;
            rowData[3] = sl.get(i).rate;
            rowData[4] = sl.get(i).amount;
            dtm.addRow(rowData);
        }
    }

    double  sum = 0;

    public void subtractValue(int row) {
        int selectedRow = row;
        DefaultTableModel model = (DefaultTableModel) table.getModel();
        sum = sum - Double.parseDouble(model.getValueAt(selectedRow,
4).toString());
        lblTotalAmount.setText("Total Amount to be paid: " + sum +
"\u20B9");
    }

    public void getTotalAmount() {
        sum = 0;
        for (int v = 0; v < table.getRowCount(); v ++) {
            sum = sum + Double.parseDouble(table.getValueAt(v,
4).toString());
        }
    }

    public void updateLabel() {
        lblTotalAmount.setText("Total Amount to be paid: " + sum + "
\u20B9");
    }
}
```

25. Products.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;

public class Products extends JFrame {

        private JPanel contentPane;

        /**
         * Launch the application.
         */
        public static void main(String[] args) {
            EventQueue.invokeLater(new Runnable() {
                public void run() {
```

```
                try {
                    Products frame = new Products();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public Products() {
        setResizable(false);
        setBounds(100, 100, 940, 173);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        contentPane.setBackground(new Color(240, 255, 255));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setLayout(null);
        panel.setBackground(new Color(240, 255, 255));
        panel.setBounds(0, 0, 1228, 79);
        contentPane.add(panel);

        JLabel lblBeautymanntraProducts = new JLabel("Beautymanntra 
Products");

    lblBeautymanntraProducts.setHorizontalAlignment(SwingConstants.CENTER);
        lblBeautymanntraProducts.setForeground(new Color(0, 124, 124));
        lblBeautymanntraProducts.setFont(new Font("Bahnschrift", Font.PLAIN, 
60));

        lblBeautymanntraProducts.setBounds(0, 0, 937, 80);
        panel.add(lblBeautymanntraProducts);

        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBackground(Color.DARK_GRAY);
        panel_1.setBounds(0, 78, 936, 67);
        contentPane.add(panel_1);

        JButton btnAddANew = new JButton("Add a new Product");
        btnAddANew.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                AddProduct a = new AddProduct();
                a.setVisible(true);
            }
        });
        btnAddANew.setForeground(Color.BLACK);
        btnAddANew.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnAddANew.setBackground(Color.WHITE);
        btnAddANew.setBounds(10, 10, 164, 35);
        panel_1.add(btnAddANew);

        JButton btnViewListOf = new JButton("View List of avalaible 
Products");
```

```
        btnViewListOf.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                DisplayTableProduct p = new DisplayTableProduct();
                p.setVisible(true);
            }
        });
        btnViewListOf.setForeground(Color.BLACK);
        btnViewListOf.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnViewListOf.setBackground(Color.WHITE);
        btnViewListOf.setBounds(184, 10, 240, 35);
        panel_1.add(btnViewListOf);

        JButton btnEditAnExisting = new JButton("Edit an existing Product");
        btnEditAnExisting.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                EditProduct ep = new EditProduct();
                ep.setVisible(true);
            }
        });
        btnEditAnExisting.setForeground(Color.BLACK);
        btnEditAnExisting.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnEditAnExisting.setBackground(Color.WHITE);
        btnEditAnExisting.setBounds(434, 10, 197, 35);
        panel_1.add(btnEditAnExisting);

        JButton btnDeleteACompany = new JButton("Delete a Product");
        btnDeleteACompany.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                DeleteProduct dp = new DeleteProduct();
                dp.setVisible(true);
            }
        });
        btnDeleteACompany.setForeground(Color.BLACK);
        btnDeleteACompany.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnDeleteACompany.setBackground(Color.WHITE);
        btnDeleteACompany.setBounds(641, 10, 167, 35);
        panel_1.add(btnDeleteACompany);

        JButton btnBack = new JButton("Back");
        btnBack.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                MainMenu mm = new MainMenu();
                mm.setVisible(true);
                dispose();
            }
        });
        btnBack.setForeground(Color.BLACK);
        btnBack.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnBack.setBackground(Color.WHITE);
        btnBack.setBounds(818, 10, 92, 35);
        panel_1.add(btnBack);
    }
}
```

26. Reports.java

```
import java.awt.BorderLayout;
import java.awt.EventQueue;
```

```java
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;
import java.awt.Color;
import javax.swing.JLabel;
import java.awt.Font;
import javax.swing.SwingConstants;
import javax.swing.JButton;
import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;

public class Reports extends JFrame {

    private JPanel contentPane;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Reports frame = new Reports();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public Reports() {
        setBounds(100, 100, 412, 181);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setLayout(null);
        panel.setBackground(new Color(240, 255, 255));
        panel.setBounds(0, 0, 482, 76);
        contentPane.add(panel);

        JLabel lblReports = new JLabel("Generate Report");
        lblReports.setHorizontalAlignment(SwingConstants.CENTER);
        lblReports.setForeground(new Color(0, 124, 124));
        lblReports.setFont(new Font("Bahnschrift", Font.PLAIN, 48));
        lblReports.setBounds(0, 0, 383, 66);
        panel.add(lblReports);

        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBackground(Color.DARK_GRAY);
        panel_1.setBounds(0, 76, 482, 68);
        contentPane.add(panel_1);
        JButton button_2 = new JButton("Back");
        button_2.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                MainMenu mm = new MainMenu();
                mm.setVisible(true);
                dispose();
            }
        });
        button_2.setForeground(Color.BLACK);
        button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_2.setBackground(Color.WHITE);
        button_2.setBounds(283, 10, 97, 35);
        panel_1.add(button_2);

        JButton btnProductInDemand = new JButton("Generate report");
        btnProductInDemand.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                Login gn = new Login();
                gn.setVisible(true);
                dispose();
            }
        });
        btnProductInDemand.setForeground(Color.BLACK);
        btnProductInDemand.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnProductInDemand.setBackground(Color.WHITE);
        btnProductInDemand.setBounds(10, 10, 226, 35);
        panel_1.add(btnProductInDemand);
    }
}
```

27. RoundedPanel.java

```
import java.awt.Color;
import java.awt.Dimension;
import java.awt.Graphics;
import java.awt.Graphics2D;
import java.awt.LayoutManager;
import java.awt.RenderingHints;

import javax.swing.JPanel;

class RoundedPanel extends JPanel
    {
        private Color backgroundColor;
        private int cornerRadius = 15;
        public RoundedPanel(LayoutManager layout, int radius) {
            super(layout);
            cornerRadius = radius;
        }
        public RoundedPanel(LayoutManager layout, int radius, Color bgColor) {
            super(layout);
            cornerRadius = radius;
            backgroundColor = bgColor;
        }
        public RoundedPanel(int radius) {
            super();
            cornerRadius = radius;
        }
        public RoundedPanel(int radius, Color bgColor) {
            super();
            cornerRadius = radius;
            backgroundColor = bgColor;
        }
        @Override
        protected void paintComponent(Graphics g) {
            super.paintComponent(g);
            Dimension arcs = new Dimension(cornerRadius, cornerRadius);
            int width = getWidth();
            int height = getHeight();
            Graphics2D graphics = (Graphics2D) g;
            graphics.setRenderingHint(RenderingHints.KEY_ANTIALIASING,
RenderingHints.VALUE_ANTIALIAS_ON);
            //Draws the rounded panel with borders.
            if (backgroundColor != null) {
                graphics.setColor(backgroundColor);
            } else {
                graphics.setColor(getBackground());
            }
            graphics.fillRoundRect(0, 0, width-1, height-1, arcs.width,
arcs.height); //paint background
            graphics.setColor(getForeground());
//            graphics.drawRoundRect(0, 0, width-1, height-1, arcs.width,
arcs.height); //paint border
//
        }
    }
```

28. Services.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.SwingConstants;
import javax.swing.border.EmptyBorder;

public class Services extends JFrame {

    private JPanel contentPane;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    Services frame = new Services();
					frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}

	/**
	 * Create the frame.
	 */
	public Services() {
		setResizable(false);
		setBounds(100, 100, 991, 181);
		contentPane = new JPanel();
		contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
		contentPane.setBackground(new Color(240, 255, 255));
		setContentPane(contentPane);
		contentPane.setLayout(null);

		JPanel panel = new JPanel();
		panel.setLayout(null);
		panel.setBorder(new EmptyBorder(5, 5, 5, 5));
		panel.setBackground(new Color(240, 255, 255));
		panel.setBounds(0, 0, 1315, 507);
		contentPane.add(panel);

		JPanel panel_1 = new JPanel();
		panel_1.setLayout(null);
		panel_1.setBorder(new EmptyBorder(5, 5, 5, 5));
		panel_1.setBackground(new Color(240, 255, 255));
		panel_1.setBounds(0, 10, 1314, 507);
		panel.add(panel_1);

		JPanel panel_2 = new JPanel();
		panel_2.setLayout(null);
		panel_2.setBackground(new Color(240, 255, 255));
		panel_2.setBounds(0, -14, 1313, 85);
		panel_1.add(panel_2);

		JLabel lblServices = new JLabel("Services");
		lblServices.setBounds(0, 0, 984, 80);
		panel_2.add(lblServices);
		lblServices.setHorizontalAlignment(SwingConstants.CENTER);
		lblServices.setForeground(new Color(0, 124, 124));
		lblServices.setFont(new Font("Bahnschrift", Font.PLAIN, 60));

		JPanel panel_3 = new JPanel();
		panel_3.setLayout(null);
		panel_3.setBackground(Color.DARK_GRAY);
		panel_3.setBounds(0, 69, 1313, 428);
		panel_1.add(panel_3);

		JButton btnAddANew = new JButton("Add a new Service");
		btnAddANew.addActionListener(new ActionListener() {
			public void actionPerformed(ActionEvent e) {
				AddService a = new AddService();
				a.setVisible(true);
			}
```

```
        });
        btnAddANew.setForeground(Color.BLACK);
        btnAddANew.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnAddANew.setBackground(Color.WHITE);
        btnAddANew.setBounds(10, 10, 191, 35);
        panel_3.add(btnAddANew);

        JButton btnViewAvailableServices = new JButton("View available
Services");
        btnViewAvailableServices.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                DisplayTableService dt = new DisplayTableService();
                dt.setVisible(true);
            }
        });
        btnViewAvailableServices.setForeground(Color.BLACK);
        btnViewAvailableServices.setFont(new Font("Bahnschrift", Font.PLAIN,
14));
        btnViewAvailableServices.setBackground(Color.WHITE);
        btnViewAvailableServices.setBounds(211, 10, 211, 35);
        panel_3.add(btnViewAvailableServices);

        JButton button_2 = new JButton("Back");
        button_2.addMouseListener(new MouseAdapter() {
            @Override
            public void mouseClicked(MouseEvent e) {
                setVisible(false);
                MainMenu mm = new MainMenu();
                mm.setVisible(true);
            }
        });
        button_2.setForeground(Color.BLACK);
        button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        button_2.setBackground(Color.WHITE);
        button_2.setBounds(881, 10, 87, 35);
        panel_3.add(button_2);

        JButton btnEditPriceOf = new JButton("Edit an Existing Service");
        btnEditPriceOf.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                EditService es = new EditService();
                es.setVisible(true);
            }
        });
        btnEditPriceOf.setForeground(Color.BLACK);
        btnEditPriceOf.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        btnEditPriceOf.setBackground(Color.WHITE);
        btnEditPriceOf.setBounds(432, 10, 264, 35);
        panel_3.add(btnEditPriceOf);

        JButton btnRemoveAService = new JButton("Remove a Service");
        btnRemoveAService.addActionListener(new ActionListener() {
            public void actionPerformed(ActionEvent e) {
                DeleteService ds = new DeleteService();
                ds.setVisible(true);
            }
        });
        btnRemoveAService.setForeground(Color.BLACK);
        btnRemoveAService.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
```

```
		btnRemoveAService.setBackground(Color.WHITE);
		btnRemoveAService.setBounds(706, 10, 165, 35);
		panel_3.add(btnRemoveAService);
	}

}
```

29. ViewClientInfo.java

```
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;

public class ViewClientInfo extends JFrame {

	private JPanel contentPane;
	protected JLabel clientLabel, nameLabel, bdayLabel, emailLabel, 
genderLabel, numberLabel;

	/**
	 * Launch the application.
	 */
	public static void main(String[] args) {
		EventQueue.invokeLater(new Runnable() {
			public void run() {
				try {
					ViewClientInfo frame = new ViewClientInfo();
					frame.setVisible(true);
				} catch (Exception e) {
					e.printStackTrace();
				}
			}
		});
	}

	/**
	 * Create the frame.
	 */
	public ViewClientInfo() {
		setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
		setBounds(100, 50, 323, 280);
		contentPane = new JPanel();
		contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
		setContentPane(contentPane);
		contentPane.setLayout(null);

		JPanel panel = new JPanel();
		panel.setBounds(0, 0, 706, 263);
		contentPane.add(panel);
		panel.setLayout(null);
```

```java
JPanel panel_1 = new JPanel();
panel_1.setLayout(null);
panel_1.setBackground(new Color(240, 255, 255));
panel_1.setBounds(0, 0, 692, 36);
panel.add(panel_1);

JLabel lblClientInfo = new JLabel("Client Info");
lblClientInfo.setForeground(new Color(0, 124, 124));
lblClientInfo.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
lblClientInfo.setBounds(0, 0, 505, 26);
panel_1.add(lblClientInfo);

JPanel panel_2 = new JPanel();
panel_2.setLayout(null);
panel_2.setBackground(Color.DARK_GRAY);
panel_2.setBounds(0, 31, 692, 218);
panel.add(panel_2);

JLabel label_1 = new JLabel("");
label_1.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
label_1.setBounds(0, 0, 0, 0);
panel_2.add(label_1);

JLabel label_2 = new JLabel("Client ID:");
label_2.setForeground(Color.WHITE);
label_2.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
label_2.setBounds(10, 23, 64, 23);
panel_2.add(label_2);

JLabel label_3 = new JLabel("Mobile Number:");
label_3.setForeground(Color.WHITE);
label_3.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
label_3.setBounds(10, 85, 96, 23);
panel_2.add(label_3);

JButton button_2 = new JButton("Back");
button_2.addActionListener(new ActionListener() {
	public void actionPerformed(ActionEvent e) {
		dispose();
	}
});
button_2.setForeground(Color.BLACK);
button_2.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
button_2.setBackground(Color.WHITE);
button_2.setBounds(109, 181, 167, 23);
panel_2.add(button_2);

JLabel label_5 = new JLabel("Email:");
label_5.setForeground(Color.WHITE);
label_5.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
label_5.setBounds(10, 116, 44, 23);
panel_2.add(label_5);

JLabel label_6 = new JLabel("Gender:");
label_6.setForeground(Color.WHITE);
label_6.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
label_6.setBounds(10, 147, 44, 23);
panel_2.add(label_6);
```

```java
        JLabel label_7 = new JLabel("Birthdate:");
        label_7.setForeground(Color.WHITE);
        label_7.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
        label_7.setBounds(111, 147, 64, 24);
        panel_2.add(label_7);

        JLabel lblName = new JLabel("Name:");
        lblName.setForeground(Color.WHITE);
        lblName.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
        lblName.setBounds(10, 54, 49, 23);
        panel_2.add(lblName);

        clientLabel = new JLabel("");
        clientLabel.setForeground(Color.GREEN);
        clientLabel.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        clientLabel.setBounds(62, 23, 214, 23);
        panel_2.add(clientLabel);

        nameLabel = new JLabel("");
        nameLabel.setForeground(Color.GREEN);
        nameLabel.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        nameLabel.setBounds(52, 54, 224, 23);
        panel_2.add(nameLabel);

        numberLabel = new JLabel("");
        numberLabel.setForeground(Color.GREEN);
        numberLabel.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        numberLabel.setBounds(100, 85, 177, 23);
        panel_2.add(numberLabel);

        emailLabel = new JLabel("");
        emailLabel.setForeground(Color.GREEN);
        emailLabel.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        emailLabel.setBounds(52, 116, 231, 23);
        panel_2.add(emailLabel);

        genderLabel = new JLabel("");
        genderLabel.setForeground(Color.GREEN);
        genderLabel.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        genderLabel.setBounds(52, 147, 59, 23);
        panel_2.add(genderLabel);

        bdayLabel = new JLabel("");
        bdayLabel.setForeground(Color.GREEN);
        bdayLabel.setFont(new Font("Bahnschrift", Font.PLAIN, 16));
        bdayLabel.setBounds(166, 147, 110, 23);
        panel_2.add(bdayLabel);
    }
}
```

30. ViewProductInfo.java

```java
import java.awt.Color;
import java.awt.EventQueue;
import java.awt.Font;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JPanel;
import javax.swing.border.EmptyBorder;

public class ViewProductInfo extends JFrame {

    private JPanel contentPane;
    JLabel pname, company, description, serialno, size, rate;

    /**
     * Launch the application.
     */
    public static void main(String[] args) {
        EventQueue.invokeLater(new Runnable() {
            public void run() {
                try {
                    ViewProductInfo frame = new ViewProductInfo();
                    frame.setVisible(true);
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });
    }

    /**
     * Create the frame.
     */
    public ViewProductInfo() {
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setBounds(100, 100, 397, 315);
        contentPane = new JPanel();
        contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));
        setContentPane(contentPane);
        contentPane.setLayout(null);

        JPanel panel = new JPanel();
        panel.setBounds(0, 0, 383, 361);
        contentPane.add(panel);
        panel.setLayout(null);

        JPanel panel_1 = new JPanel();
        panel_1.setLayout(null);
        panel_1.setBackground(new Color(240, 255, 255));
        panel_1.setBounds(0, 0, 383, 36);
        panel.add(panel_1);

        JLabel lblProductInfo = new JLabel("Product Info");
        lblProductInfo.setForeground(new Color(0, 124, 124));
        lblProductInfo.setFont(new Font("Bahnschrift", Font.PLAIN, 24));
        lblProductInfo.setBounds(0, 0, 383, 26);
        panel_1.add(lblProductInfo);

        JPanel panel_2 = new JPanel();
        panel_2.setLayout(null);
        panel_2.setBackground(Color.DARK_GRAY);
        panel_2.setBounds(0, 36, 383, 244);
        panel.add(panel_2);
```

```
             JLabel lblSerialNumber = new JLabel("Serial Number: ");
             lblSerialNumber.setForeground(Color.WHITE);
             lblSerialNumber.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
             lblSerialNumber.setBounds(10, 10, 83, 45);
             panel_2.add(lblSerialNumber);

             JLabel lblProductName = new JLabel("Product Name:");
             lblProductName.setForeground(Color.WHITE);
             lblProductName.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
             lblProductName.setBounds(10, 78, 78, 45);
             panel_2.add(lblProductName);

             JButton button = new JButton("Back");
             button.addActionListener(new ActionListener() {
                    public void actionPerformed(ActionEvent e) {
                           dispose();
                    }
             });
             button.setForeground(Color.BLACK);
             button.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
             button.setBackground(Color.WHITE);
             button.setBounds(110, 210, 167, 23);
             panel_2.add(button);

             JLabel lblProductDescription = new JLabel("<HTML>Product
Description:<HTML>");
             lblProductDescription.setForeground(Color.WHITE);
             lblProductDescription.setFont(new Font("Bahnschrift", Font.PLAIN,
12));
             lblProductDescription.setBounds(10, 116, 75, 51);
             panel_2.add(lblProductDescription);

             JLabel lblSize = new JLabel("Size:");
             lblSize.setForeground(Color.WHITE);
             lblSize.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
             lblSize.setBounds(10, 177, 25, 15);
             panel_2.add(lblSize);

             JLabel lblRate = new JLabel("Rate: ");
             lblRate.setForeground(Color.WHITE);
             lblRate.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
             lblRate.setBounds(208, 177, 30, 15);
             panel_2.add(lblRate);

             JLabel lblCompany = new JLabel("Company:");
             lblCompany.setForeground(Color.WHITE);
             lblCompany.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
             lblCompany.setBounds(10, 40, 51, 45);
             panel_2.add(lblCompany);

             serialno = new JLabel("");
             serialno.setForeground(Color.GREEN);
             serialno.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
             serialno.setBounds(103, 10, 214, 45);
             panel_2.add(serialno);

             company = new JLabel("");
             company.setForeground(Color.GREEN);
```

```java
        company.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        company.setBounds(74, 40, 299, 45);
        panel_2.add(company);

        pname = new JLabel("<HTML>" + "<HTML>");
        pname.setForeground(Color.GREEN);
        pname.setFont(new Font("Bahnschrift", Font.PLAIN, 12));
        pname.setBounds(98, 78, 275, 45);
        panel_2.add(pname);

        description = new JLabel("<HTML>" + "<HTML>");
        description.setForeground(Color.GREEN);
        description.setFont(new Font("Bahnschrift", Font.PLAIN, 10));
        description.setBounds(85, 116, 288, 51);
        panel_2.add(description);

        size = new JLabel("");
        size.setForeground(Color.GREEN);
        size.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        size.setBounds(41, 169, 83, 31);
        panel_2.add(size);

        rate = new JLabel("");
        rate.setForeground(Color.GREEN);
        rate.setFont(new Font("Bahnschrift", Font.PLAIN, 14));
        rate.setBounds(248, 169, 125, 31);
        panel_2.add(rate);
    }
}
```