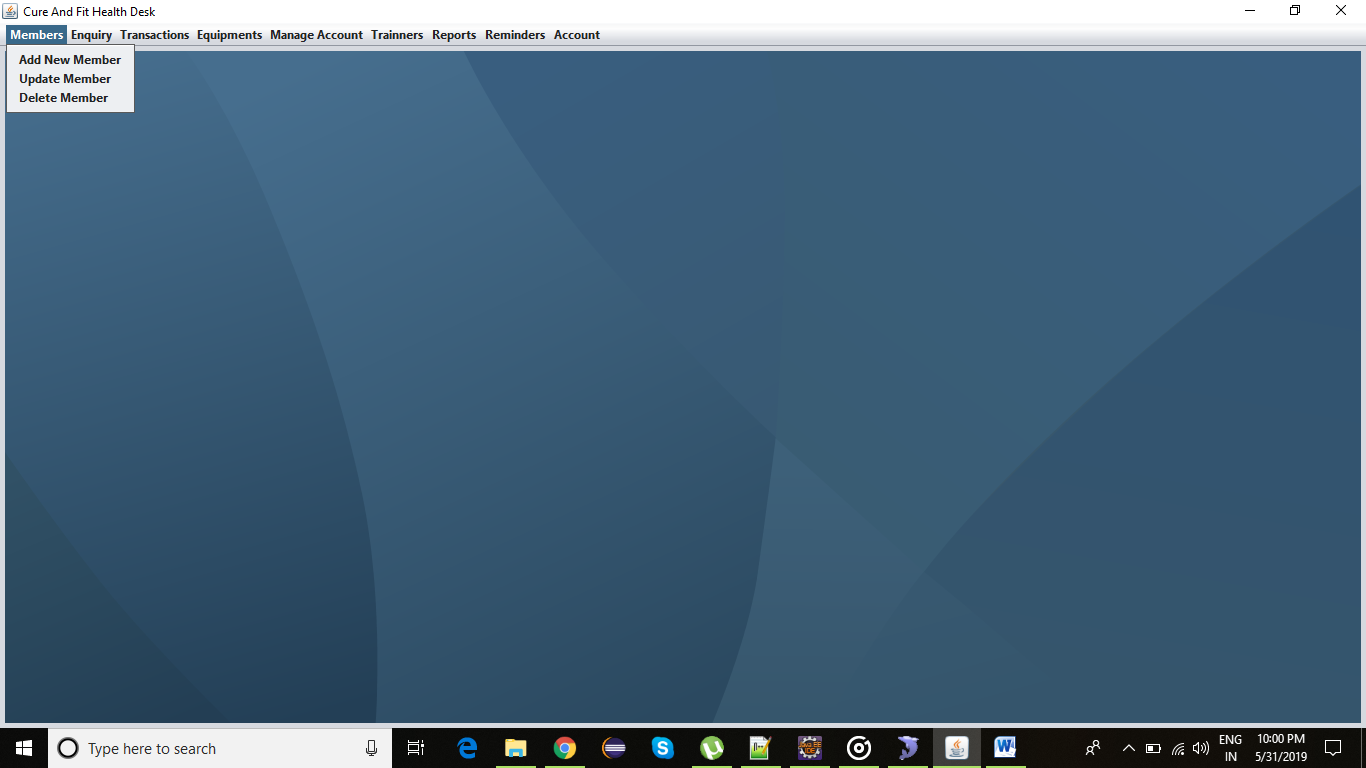
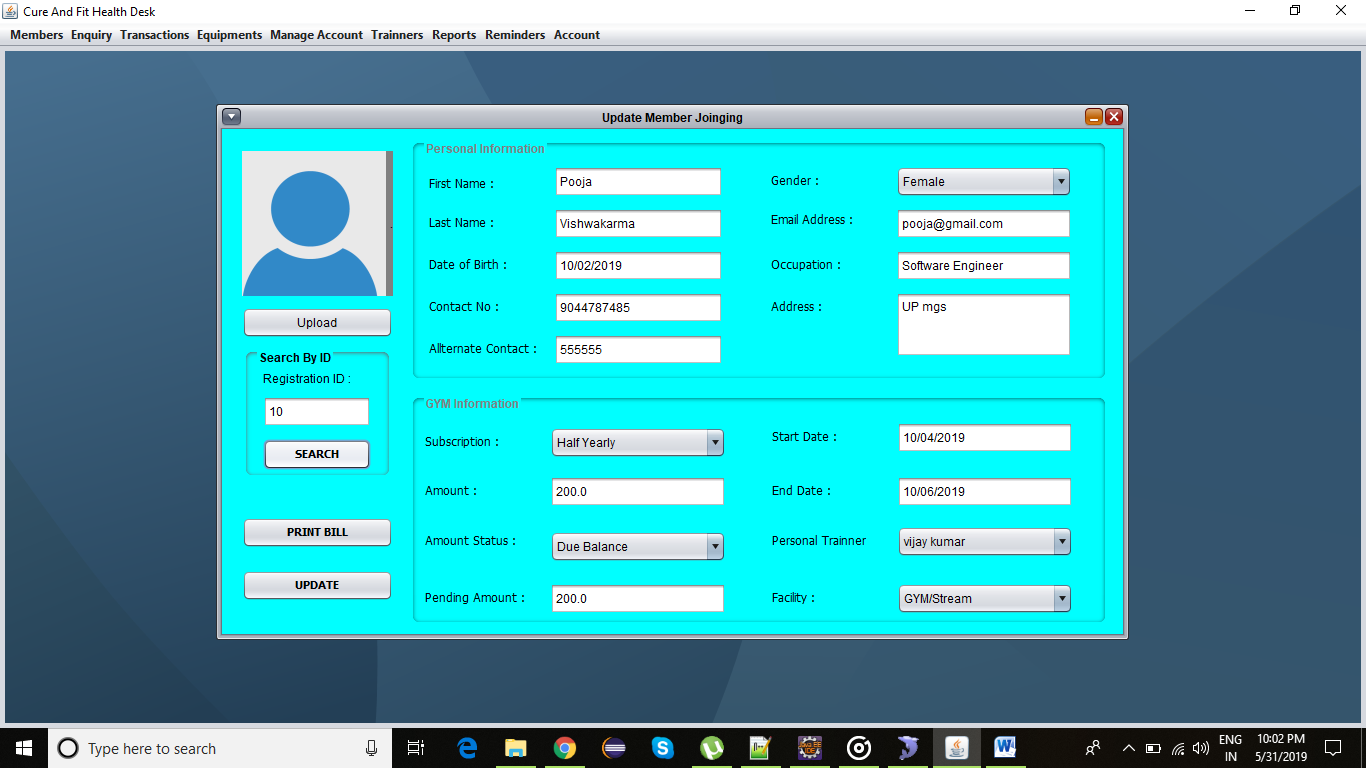
**SCREENS**

**Desktop View**

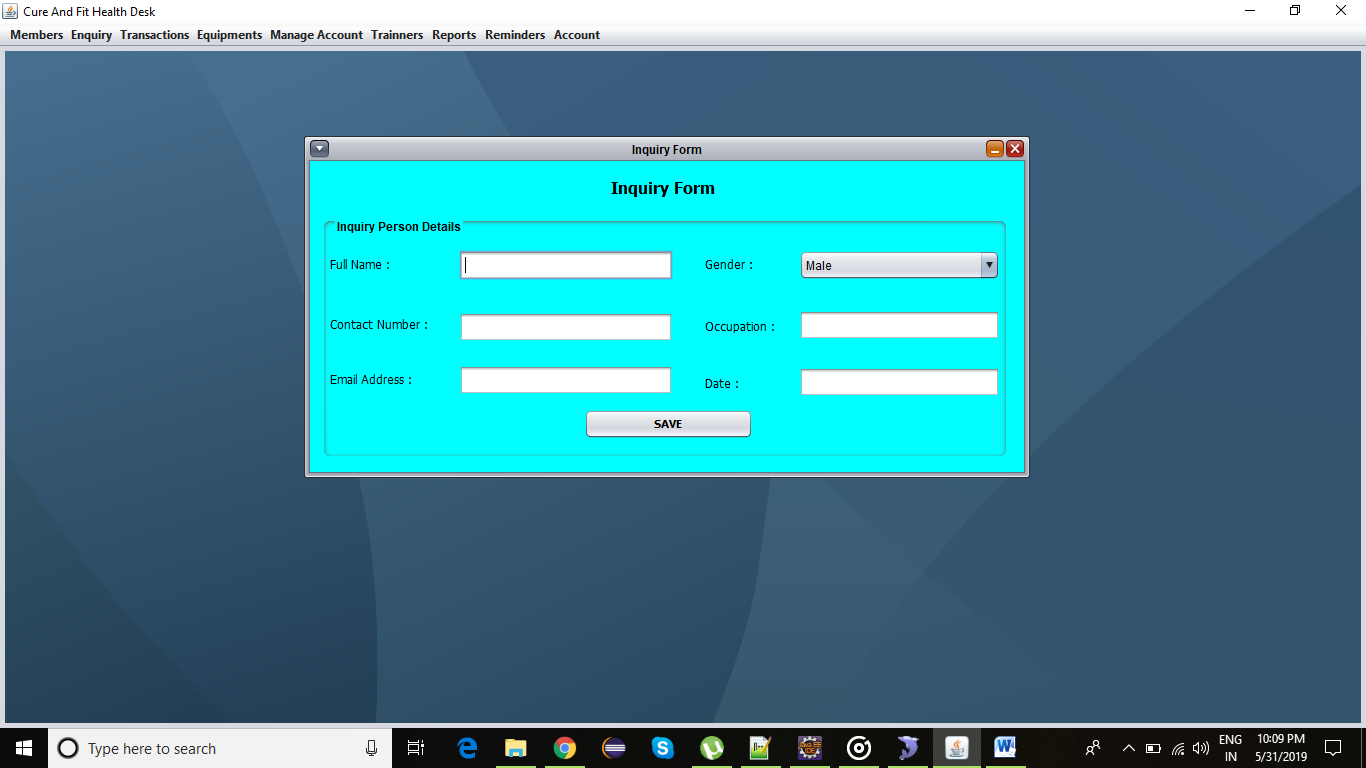
**Update Member Joining View**



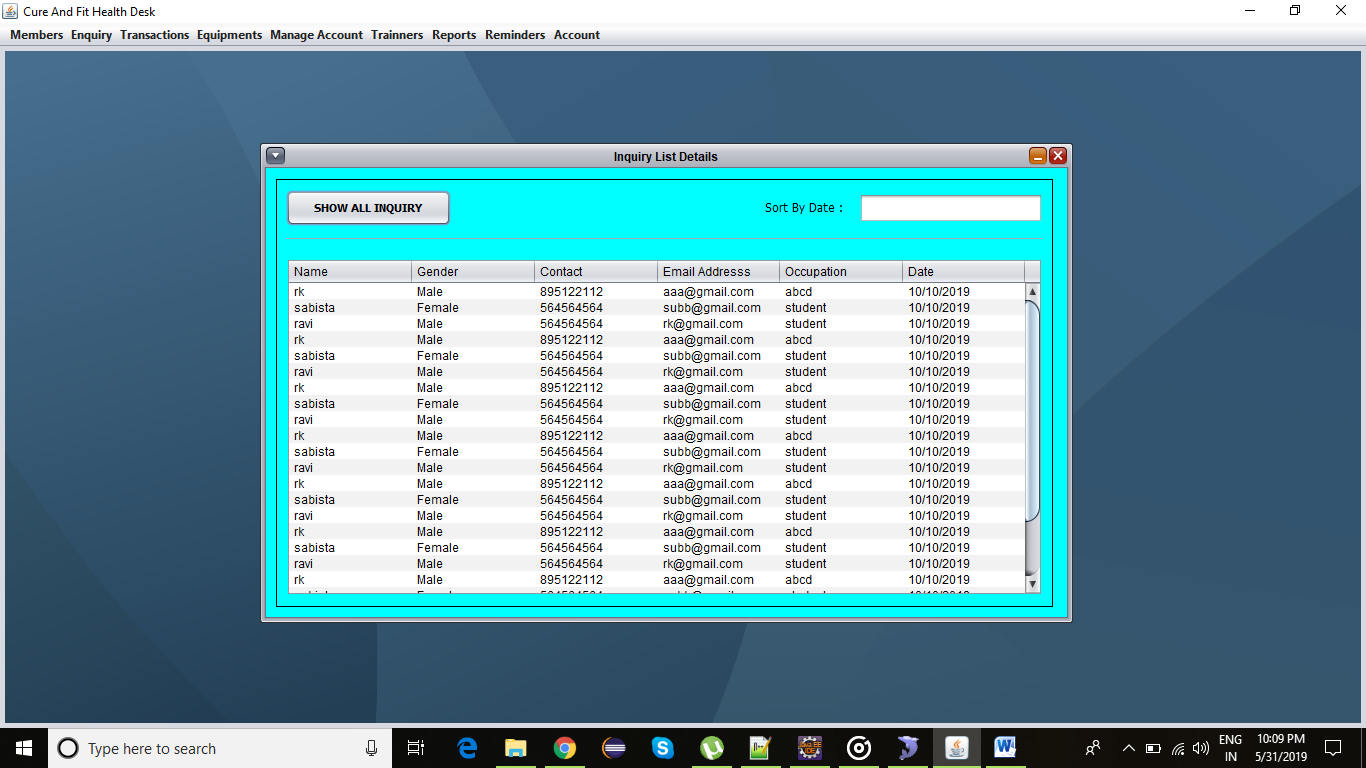
**New Member Joining View**



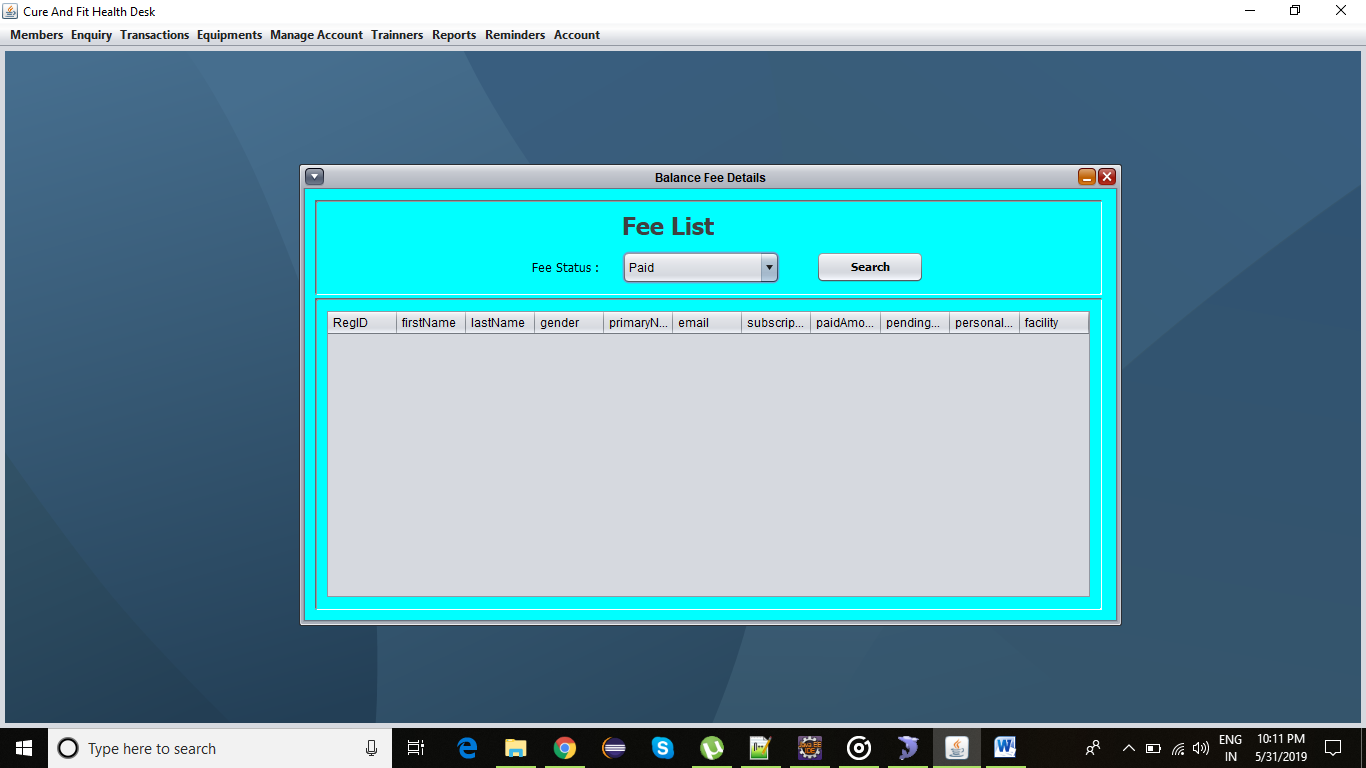
**Inquiry Form View**



**Inquiry List Details View**



**Balance Fee List Details View**



**cODES**

**FitnessDesk.java**

package com.cure.fit.main;

import java.awt.BorderLayout;

import java.awt.EventQueue;

import java.awt.Font;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JDesktopPane;

import javax.swing.JFrame;

import javax.swing.JMenu;

import javax.swing.JMenuBar;

import javax.swing.JMenuItem;

import javax.swing.JPanel;

import javax.swing.UIManager;

import javax.swing.UIManager.LookAndFeelInfo;

import javax.swing.border.EmptyBorder;

import com.cure.fit.inquiry.view.InquiryForm;

import com.cure.fit.inquiry.view.InquiryListView;

import com.cure.fit.member.view.JoinginNewMemberView;

import com.cure.fit.member.view.UpdateMemberDetailsView;

import com.cure.fit.transaction.view.BalanceFeesView;

public class FitnessDesk extends JFrame {

/\*\*

\* serialVersionUID

\*/

private static final long serialVersionUID = -5787145458966091501L;

private JPanel contentPane;

private JDesktopPane desktopPane;

private static FitnessDesk frame;

private static JoinginNewMemberView newJoiningMemberView;

private static UpdateMemberDetailsView updateMemberView;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

try {

for (LookAndFeelInfo info : UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (Exception e) {

// If Nimbus is not available, you can set the GUI to

// another look and feel.

}

frame = new FitnessDesk();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public FitnessDesk() {

setTitle("Cure And Fit Health Desk");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setBounds(100, 100, 782, 465);

JMenuBar menuBar = new JMenuBar();

setJMenuBar(menuBar);

JMenu mnMembers = new JMenu("Members");

mnMembers.setFont(new Font("Segoe UI", Font.BOLD, 12));

menuBar.add(mnMembers);

JMenuItem mntmAddNewMember = new JMenuItem("Add New Member");

mntmAddNewMember.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

newJoiningMemberView = new JoinginNewMemberView();

desktopPane.add(newJoiningMemberView);

newJoiningMemberView.show();

}

});

mntmAddNewMember.setFont(new Font("Segoe UI", Font.BOLD, 12));

mnMembers.add(mntmAddNewMember);

JMenuItem mntmUpdateMember = new JMenuItem("Update Member");

mntmUpdateMember.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

updateMemberView = new UpdateMemberDetailsView();

desktopPane.add(updateMemberView);

updateMemberView.show();

}

});

mntmUpdateMember.setFont(new Font("Segoe UI", Font.BOLD, 12));

mnMembers.add(mntmUpdateMember);

JMenuItem mntmDeleteMember = new JMenuItem("Delete Member");

mntmDeleteMember.setFont(new Font("Segoe UI", Font.BOLD, 12));

mnMembers.add(mntmDeleteMember);

JMenu mnEnquiry = new JMenu("Enquiry");

mnEnquiry.setFont(new Font("Segoe UI", Font.BOLD, 12));

menuBar.add(mnEnquiry);

JMenuItem mntmEnquiryForm = new JMenuItem("Enquiry Form");

mntmEnquiryForm.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

InquiryForm inqFrm=new InquiryForm();

desktopPane.add(inqFrm);

inqFrm.show();

}

});

mntmEnquiryForm.setFont(new Font("Segoe UI", Font.BOLD, 12));

mnEnquiry.add(mntmEnquiryForm);

JMenuItem mntmViewEnquiryList = new JMenuItem("View Enquiry List");

mntmViewEnquiryList.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

InquiryListView inquiryList=new InquiryListView();

desktopPane.add(inquiryList);

inquiryList.show();

}

});

mntmViewEnquiryList.setFont(new Font("Segoe UI", Font.BOLD, 12));

mnEnquiry.add(mntmViewEnquiryList);

JMenu mnTransactions = new JMenu("Transactions");

mnTransactions.setFont(new Font("Segoe UI", Font.BOLD, 12));

menuBar.add(mnTransactions);

JMenuItem mntmBalance = new JMenuItem("Balance Fees");

mntmBalance.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

BalanceFeesView blcFeeView=new BalanceFeesView();

desktopPane.add(blcFeeView);

blcFeeView.show();

}

});

mnTransactions.add(mntmBalance);

JMenu mnEquipments = new JMenu("Equipments");

mnEquipments.setFont(new Font("Segoe UI", Font.BOLD, 12));

menuBar.add(mnEquipments);

JMenu mnManageAccount = new JMenu("Manage Account");

mnManageAccount.setFont(new Font("Segoe UI", Font.BOLD, 12));

menuBar.add(mnManageAccount);

JMenu mnTrainners = new JMenu("Trainners");

mnTrainners.setFont(new Font("Segoe UI", Font.BOLD, 12));

menuBar.add(mnTrainners);

JMenu mnReports = new JMenu("Reports");

mnReports.setFont(new Font("Segoe UI", Font.BOLD, 12));

menuBar.add(mnReports);

JMenu mnReminders = new JMenu("Reminders");

mnReminders.setFont(new Font("Segoe UI", Font.BOLD, 12));

menuBar.add(mnReminders);

JMenu mnAccount = new JMenu("Account");

mnAccount.setFont(new Font("Segoe UI", Font.BOLD, 12));

menuBar.add(mnAccount);

contentPane = new JPanel();

contentPane.setBorder(new EmptyBorder(5, 5, 5, 5));

contentPane.setLayout(new BorderLayout(0, 0));

setContentPane(contentPane);

desktopPane = new JDesktopPane();

contentPane.add(desktopPane, BorderLayout.CENTER);

}

public static JoinginNewMemberView getView()

{

return newJoiningMemberView;

}

}

**InquiryDao.java**

package com.cure.fit.inquiry.dao;

import java.sql.Connection;

import java.util.List;

import com.cure.fit.inquiry.model.InquryDetialsVO;

import com.cure.fit.util.MysqlConnectionUtil;

public interface InquiryDao {

Connection con=MysqlConnectionUtil.createConnection();

boolean saveInquiryDetails(InquryDetialsVO vo);

boolean updateInquiryDetails(InquryDetialsVO vo);

List<InquryDetialsVO> selectAllInquiryMemberDetails();

}

**InquiryDaoImpl.java**

package com.cure.fit.inquiry.dao;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

import java.util.List;

import com.cure.fit.inquiry.model.InquryDetialsVO;

public class InquiryDaoImpl implements InquiryDao {

private static InquiryDaoImpl inqDts;

@Override

public boolean saveInquiryDetails(InquryDetialsVO vo) {

try {

PreparedStatement st = con.prepareStatement("insert into inquirydetails values(?,?,?,?,?,?)");

st.setString(1, vo.getFullName());

st.setLong(2, vo.getContactNumber());

st.setString(3, vo.getEmail());

st.setString(4, vo.getGender());

st.setString(5, vo.getOccupation());

st.setString(6, vo.getDate());

if (st.executeUpdate() > 0) {

return true;

}

} catch (SQLException e) {

e.printStackTrace();

return false;

}

return false;

}

@Override

public boolean updateInquiryDetails(InquryDetialsVO vo) {

return false;

}

public static InquiryDaoImpl getInstance() {

if (inqDts == null) {

inqDts = new InquiryDaoImpl();

}

return inqDts;

}

@Override

public List<InquryDetialsVO> selectAllInquiryMemberDetails() {

try {

PreparedStatement st = con.prepareStatement("select \* from inquirydetails");

ResultSet rs=st.executeQuery();

List<InquryDetialsVO> inquiryList=new ArrayList<>();

if(rs!=null)

{

while(rs.next())

{

InquryDetialsVO vo=new InquryDetialsVO();

vo.setFullName(rs.getString(1));

vo.setContactNumber(rs.getLong(2));

vo.setEmail(rs.getString(3));

vo.setGender(rs.getString(4));

vo.setOccupation(rs.getString(5));

vo.setDate(rs.getString(6));

inquiryList.add(vo);

}

}

System.out.println(inquiryList);

return inquiryList;

} catch (SQLException e) {

e.printStackTrace();

}

return null;

}

}

**InquryDetialsVO.java**

package com.cure.fit.inquiry.model;

public class InquryDetialsVO {

private String fullName;

private Long contactNumber;

private String email;

private String gender;

private String occupation;

private String date;

public String getFullName() {

return fullName;

}

public void setFullName(String fullName) {

this.fullName = fullName;

}

public Long getContactNumber() {

return contactNumber;

}

public void setContactNumber(Long contactNumber) {

this.contactNumber = contactNumber;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getGender() {

return gender;

}

public void setGender(String gender) {

this.gender = gender;

}

public String getOccupation() {

return occupation;

}

public void setOccupation(String occupation) {

this.occupation = occupation;

}

public String getDate() {

return date;

}

public void setDate(String date) {

this.date = date;

}

@Override

public String toString() {

return "InquryDetialsVO [fullName=" + fullName + ", contactNumber=" + contactNumber + ", email=" + email

+ ", gender=" + gender + ", occupation=" + occupation + ", date=" + date + "]";

}

}

**InquiryForm.java**

package com.cure.fit.inquiry.view;

import java.awt.EventQueue;

import javax.swing.JInternalFrame;

import java.awt.Color;

import javax.swing.JPanel;

import java.awt.BorderLayout;

import javax.swing.UIManager;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import java.awt.Font;

import javax.swing.border.TitledBorder;

import com.cure.fit.inquiry.dao.InquiryDao;

import com.cure.fit.inquiry.dao.InquiryDaoImpl;

import com.cure.fit.inquiry.model.InquryDetialsVO;

import com.cure.fit.main.FitnessDesk;

import javax.swing.JTextField;

import javax.swing.JComboBox;

import javax.swing.JButton;

import javax.swing.DefaultComboBoxModel;

import java.awt.event.ActionListener;

import java.awt.event.ActionEvent;

public class InquiryForm extends JInternalFrame {

private JTextField textField;

private JTextField textField\_1;

private JTextField textField\_2;

private JTextField textField\_3;

private JTextField textField\_4;

private JComboBox comboBox;

private InquryDetialsVO inqDetailsVO;

private InquiryDao dao = new InquiryDaoImpl();

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

InquiryForm frame = new InquiryForm();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public InquiryForm() {

setTitle("Inquiry Form");

setClosable(true);

setIconifiable(true);

getContentPane().setBackground(Color.CYAN);

JPanel panel = new JPanel();

panel.setBackground(UIManager.getColor("InternalFrame.activeTitleBackground"));

getContentPane().add(panel, BorderLayout.CENTER);

panel.setLayout(null);

JLabel lblInquiryForm = new JLabel("Inquiry Form");

lblInquiryForm.setForeground(Color.BLACK);

lblInquiryForm.setFont(new Font("Tahoma", Font.BOLD, 16));

lblInquiryForm.setBounds(301, 9, 149, 37);

panel.add(lblInquiryForm);

JPanel panel\_1 = new JPanel();

panel\_1.setOpaque(false);

panel\_1.setBorder(new TitledBorder(null, "Inquiry Person Details", TitledBorder.LEADING, TitledBorder.TOP, null, null));

panel\_1.setBounds(10, 57, 690, 244);

panel.add(panel\_1);

panel\_1.setLayout(null);

JLabel lblFullName = new JLabel("Full Name :");

lblFullName.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblFullName.setBounds(10, 34, 96, 24);

panel\_1.add(lblFullName);

textField = new JTextField();

textField.setBounds(139, 32, 214, 30);

panel\_1.add(textField);

textField.setColumns(10);

JLabel lblContactNumber = new JLabel("Contact Number :");

lblContactNumber.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblContactNumber.setBounds(10, 94, 143, 24);

panel\_1.add(lblContactNumber);

textField\_1 = new JTextField();

textField\_1.setColumns(10);

textField\_1.setBounds(139, 94, 214, 30);

panel\_1.add(textField\_1);

JLabel lblGender = new JLabel("Gender :");

lblGender.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblGender.setBounds(385, 34, 85, 24);

panel\_1.add(lblGender);

comboBox = new JComboBox();

comboBox.setModel(new DefaultComboBoxModel(new String[] { "Male", "Female", "Other" }));

comboBox.setBounds(479, 32, 201, 30);

panel\_1.add(comboBox);

JLabel lblEmail = new JLabel("Email Address :");

lblEmail.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblEmail.setBounds(10, 149, 85, 24);

panel\_1.add(lblEmail);

textField\_2 = new JTextField();

textField\_2.setColumns(10);

textField\_2.setBounds(139, 147, 214, 30);

panel\_1.add(textField\_2);

JLabel lblOccupation = new JLabel("Occupation :");

lblOccupation.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblOccupation.setBounds(385, 96, 85, 24);

panel\_1.add(lblOccupation);

textField\_3 = new JTextField();

textField\_3.setColumns(10);

textField\_3.setBounds(479, 92, 201, 30);

panel\_1.add(textField\_3);

JLabel lblDate = new JLabel("Date :");

lblDate.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblDate.setBounds(385, 153, 85, 24);

panel\_1.add(lblDate);

textField\_4 = new JTextField();

textField\_4.setColumns(10);

textField\_4.setBounds(479, 149, 201, 30);

panel\_1.add(textField\_4);

JButton btnNewButton = new JButton("SAVE");

btnNewButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

loadInquiryFormData();

boolean saevStatus = dao.saveInquiryDetails(inqDetailsVO);

if (saevStatus) {

JOptionPane.showMessageDialog(FitnessDesk.getView(), "The record is saved successfuly");

} else {

JOptionPane.showMessageDialog(FitnessDesk.getView(), "The record is not saved successfuly");

}

}

});

btnNewButton.setFont(new Font("Tahoma", Font.BOLD, 11));

btnNewButton.setBounds(264, 191, 169, 30);

panel\_1.add(btnNewButton);

setBounds(100, 100, 726, 342);

}

private void loadInquiryFormData() {

inqDetailsVO = new InquryDetialsVO();

inqDetailsVO.setFullName(textField.getText());

inqDetailsVO.setContactNumber(Long.parseLong(textField\_1.getText()));

inqDetailsVO.setDate(textField\_4.getText());

inqDetailsVO.setEmail(textField\_2.getText());

inqDetailsVO.setGender(comboBox.getSelectedItem().toString());

inqDetailsVO.setOccupation(textField\_3.getText());

}

}

**InquiryListView.java**

package com.cure.fit.inquiry.view;

import java.awt.EventQueue;

import javax.swing.JInternalFrame;

import javax.swing.JPanel;

import java.awt.Color;

import javax.swing.JButton;

import javax.swing.JSeparator;

import javax.swing.JLabel;

import java.awt.Font;

import javax.swing.JTextField;

import javax.swing.JTable;

import javax.swing.table.DefaultTableModel;

import com.cure.fit.inquiry.dao.InquiryDao;

import com.cure.fit.inquiry.dao.InquiryDaoImpl;

import com.cure.fit.inquiry.model.InquryDetialsVO;

import javax.swing.ListSelectionModel;

import javax.swing.JScrollPane;

import javax.swing.border.LineBorder;

import java.awt.event.ActionListener;

import java.util.List;

import java.awt.event.ActionEvent;

public class InquiryListView extends JInternalFrame {

private JTextField textField;

private JTable table;

private InquiryDao dao = new InquiryDaoImpl();

DefaultTableModel model;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

InquiryListView frame = new InquiryListView();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public InquiryListView() {

setTitle("Inquiry List Details");

setClosable(true);

setIconifiable(true);

getContentPane().setBackground(Color.CYAN);

setBounds(100, 100, 813, 480);

getContentPane().setLayout(null);

JPanel panel = new JPanel();

panel.setOpaque(false);

panel.setBorder(new LineBorder(Color.BLACK));

panel.setBounds(10, 11, 777, 428);

getContentPane().add(panel);

panel.setLayout(null);

JButton btnShowAllInquiry = new JButton("SHOW ALL INQUIRY");

btnShowAllInquiry.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

List<InquryDetialsVO> inqDetialsList = dao.selectAllInquiryMemberDetails();

loadDataInTable(inqDetialsList);

}

});

btnShowAllInquiry.setBounds(10, 11, 165, 36);

btnShowAllInquiry.setFont(new Font("Tahoma", Font.BOLD, 11));

panel.add(btnShowAllInquiry);

JSeparator separator = new JSeparator();

separator.setBounds(10, 58, 757, 10);

panel.add(separator);

JLabel lblSortByDate = new JLabel("Sort By Date :");

lblSortByDate.setBounds(489, 13, 84, 30);

lblSortByDate.setFont(new Font("Tahoma", Font.PLAIN, 12));

panel.add(lblSortByDate);

textField = new JTextField();

textField.setBounds(583, 14, 184, 30);

panel.add(textField);

textField.setColumns(10);

JScrollPane scrollPane = new JScrollPane();

scrollPane.setBounds(10, 79, 757, 338);

panel.add(scrollPane);

table = new JTable();

model = new DefaultTableModel(

new String[] { "Name", "Gender", "Contact", "Email Addresss", "Occupation" ,"Date" }, 0);

table.setModel(model);

scrollPane.setViewportView(table);

}

private void loadDataInTable(List<InquryDetialsVO> list) {

for (InquryDetialsVO vo : list) {

Object[] rowDetails = {vo.getFullName(), vo.getGender(), vo.getContactNumber(),vo.getEmail(),vo.getOccupation(),vo.getDate()};

model.addRow(rowDetails);

}

}

}

**DBOperation.java**

package com.cure.fit.member.dao;

import java.sql.Connection;

import com.cure.fit.members.model.MemberGymInfoVO;

import com.cure.fit.members.model.MemberPersonalInfoVO;

import com.cure.fit.util.MysqlConnectionUtil;

public interface DBOperation {

Connection con=MysqlConnectionUtil.createConnection();

boolean save(MemberPersonalInfoVO vo1,MemberGymInfoVO vo2);

boolean update(MemberPersonalInfoVO vo1,MemberGymInfoVO vo2);

boolean detete();

}

**JoiningNewMemberDAOImpl.java**

package com.cure.fit.member.dao;

import java.io.File;

import java.io.FileInputStream;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.List;

import java.util.Vector;

import javax.swing.JOptionPane;

import com.cure.fit.main.FitnessDesk;

import com.cure.fit.members.model.MemberGymInfoVO;

import com.cure.fit.members.model.MemberInfo;

import com.cure.fit.members.model.MemberPersonalInfoVO;

import com.mysql.jdbc.exceptions.MySQLIntegrityConstraintViolationException;

public class JoiningNewMemberDAOImpl implements DBOperation {

private static JoiningNewMemberDAOImpl newMember;

@Override

public boolean save(MemberPersonalInfoVO vo1, MemberGymInfoVO vo2) {

try {

con.setAutoCommit(false);

PreparedStatement ps\_perInfo = con

.prepareStatement("insert into newmemberinfo values(?,?,?,?,?,?,?,?,?,?,?)");

PreparedStatement ps\_perGYMInfo = con

.prepareStatement("insert into membergyminfo values(?,?,?,?,?,?,?,?,?)");

ps\_perInfo.setLong(1, (vo1.getRegistrationID()));

ps\_perInfo.setString(2, vo1.getFirstName());

ps\_perInfo.setString(3, vo1.getLastName());

ps\_perInfo.setString(4, vo1.getDob());

ps\_perInfo.setLong(5, vo1.getPrimaryNumber());

ps\_perInfo.setLong(6, vo1.getAlternetNumber());

ps\_perInfo.setString(7, vo1.getGender());

ps\_perInfo.setString(8, vo1.getEmail());

ps\_perInfo.setString(9, vo1.getOccupation());

ps\_perInfo.setString(10, vo1.getAddress());

File file = new File(vo1.getImagePath());

ps\_perInfo.setBinaryStream(11, new FileInputStream(file), (int) file.length());

ps\_perGYMInfo.setLong(1, vo2.getRegistrationID());

ps\_perGYMInfo.setString(2, vo2.getSubscription());

ps\_perGYMInfo.setDouble(3, vo2.getPaidAmount());

ps\_perGYMInfo.setDouble(4, vo2.getPendingAmount());

ps\_perGYMInfo.setString(5, vo2.getStatus());

ps\_perGYMInfo.setString(6, vo2.getStartDate());

ps\_perGYMInfo.setString(7, vo2.getEndDate());

ps\_perGYMInfo.setString(8, vo2.getPersonalTranner());

ps\_perGYMInfo.setString(9, vo2.getFacility());

if (ps\_perGYMInfo.executeUpdate() > 0 && ps\_perInfo.executeUpdate() > 0) {

con.commit();

return true;

}

} catch (MySQLIntegrityConstraintViolationException e) {

try {

con.rollback();

JOptionPane.showMessageDialog(FitnessDesk.getView(), "This record is already saved..");

} catch (SQLException e1) {

System.out.println("Error in Rollback new member joing dao");

}

e.printStackTrace();

} catch (Exception e) {

try {

con.rollback();

} catch (SQLException e1) {

System.out.println("Error in Rollback new member joing dao!");

}

e.printStackTrace();

}

return false;

}

public List<MemberInfo> searchMemberByID(Long registrationId) {

MemberInfo vo1 = new MemberPersonalInfoVO();

MemberInfo vo2 = new MemberGymInfoVO();

List<MemberInfo> ls\_Member = new ArrayList<>();

final String query1 = "select \* from newmemberinfo where regId = " + registrationId;

final String query2 = "select \* from membergyminfo where regId = " + registrationId;

try {

Statement st1 = con.createStatement();

Statement st2 = con.createStatement();

ResultSet rs1 = st1.executeQuery(query1);

ResultSet rs2 = st2.executeQuery(query2);

while (rs1.next()) {

((MemberPersonalInfoVO) vo1).setRegistrationID(rs1.getLong(1));

((MemberPersonalInfoVO) vo1).setFirstName(rs1.getString(2));

((MemberPersonalInfoVO) vo1).setLastName(rs1.getString(3));

((MemberPersonalInfoVO) vo1).setDob(rs1.getString(4));

((MemberPersonalInfoVO) vo1).setPrimaryNumber(rs1.getLong(5));

((MemberPersonalInfoVO) vo1).setAlternetNumber(rs1.getLong(6));

((MemberPersonalInfoVO) vo1).setGender(rs1.getString(7));

((MemberPersonalInfoVO) vo1).setEmail(rs1.getString(8));

((MemberPersonalInfoVO) vo1).setOccupation(rs1.getString(9));

((MemberPersonalInfoVO) vo1).setAddress(rs1.getString(10));

}

while (rs2.next()) {

((MemberGymInfoVO) vo2).setRegistrationID(rs2.getLong(1));

((MemberGymInfoVO) vo2).setSubscription(rs2.getString(2));

((MemberGymInfoVO) vo2).setPaidAmount(rs2.getDouble(3));

((MemberGymInfoVO) vo2).setPendingAmount(rs2.getDouble(4));

((MemberGymInfoVO) vo2).setStatus(rs2.getString(5));

((MemberGymInfoVO) vo2).setStartDate(rs2.getString(6));

((MemberGymInfoVO) vo2).setEndDate(rs2.getString(7));

((MemberGymInfoVO) vo2).setPersonalTranner(rs2.getString(8));

((MemberGymInfoVO) vo2).setFacility(rs2.getString(9));

}

ls\_Member.add(vo1);

ls\_Member.add(vo2);

} catch (SQLException e) {

e.printStackTrace();

}

return ls\_Member;

}

@Override

public boolean detete() {

// TODO Auto-generated method stub

return false;

}

/\*\*

\* @return list of trainer name

\*/

public Vector<String> getTrainnerName() {

Vector<String> listTrannerNames = new Vector<>();

try {

Statement stmt = con.createStatement();

ResultSet rs = stmt.executeQuery("select fname,lname from trainnerinfo");

listTrannerNames.add("None");

while (rs.next()) {

listTrannerNames.add(

new StringBuilder().append(rs.getString(1)).append(" ").append(rs.getString(2)).toString());

}

} catch (Exception e) {

System.out.println(e);

}

return listTrannerNames;

}

public int getNextGeneratedID() {

try {

Statement stmt = con.createStatement();

ResultSet rs = stmt.executeQuery("select max(regId) from newmemberinfo");

if (rs.next()) {

System.out.println(rs.getInt(1));

return rs.getInt(1) + 1;

}

} catch (Exception ex) {

ex.printStackTrace();

}

return 0;

}

public static JoiningNewMemberDAOImpl getInstance() {

if (newMember == null) {

newMember = new JoiningNewMemberDAOImpl();

}

return newMember;

}

@Override

public boolean update(MemberPersonalInfoVO vo1, MemberGymInfoVO vo2) {

try {

con.setAutoCommit(false);

PreparedStatement ps\_perInfo = con.prepareStatement(

"update newmemberinfo set regId=?,fname=?,lname=?,dob=?,contact=?,alternate\_contact=?,gender=?,email=?,occupation=?,address=?,photo=? where regId=?");

PreparedStatement ps\_perGYMInfo = con.prepareStatement(

"update membergyminfo set regId=?,subscription=?,paidAmount=?,pendingAmount=?,status=?,startDate=?,endDate=?,personalTranner=?,facility=? where regId=?");

ps\_perInfo.setLong(1, (vo1.getRegistrationID()));

ps\_perInfo.setString(2, vo1.getFirstName());

ps\_perInfo.setString(3, vo1.getLastName());

ps\_perInfo.setString(4, vo1.getDob());

ps\_perInfo.setLong(5, vo1.getPrimaryNumber());

ps\_perInfo.setLong(6, vo1.getAlternetNumber());

ps\_perInfo.setString(7, vo1.getGender());

ps\_perInfo.setString(8, vo1.getEmail());

ps\_perInfo.setString(9, vo1.getOccupation());

ps\_perInfo.setString(10, vo1.getAddress());

if(vo1.getImagePath()==null ||vo1.getImagePath().equals(""))

{

String defualtImgPath = "./images/com/cure/fit/profile/defaultUserPic.jpg";

vo1.setImagePath(defualtImgPath);

}

File file = new File(vo1.getImagePath());

ps\_perInfo.setBinaryStream(11, new FileInputStream(file), (int) file.length());

ps\_perInfo.setLong(12, vo1.getRegistrationID());

ps\_perGYMInfo.setLong(1, vo2.getRegistrationID());

ps\_perGYMInfo.setString(2, vo2.getSubscription());

ps\_perGYMInfo.setDouble(3, vo2.getPaidAmount());

ps\_perGYMInfo.setDouble(4, vo2.getPendingAmount());

ps\_perGYMInfo.setString(5, vo2.getStatus());

ps\_perGYMInfo.setString(6, vo2.getStartDate());

ps\_perGYMInfo.setString(7, vo2.getEndDate());

ps\_perGYMInfo.setString(8, vo2.getPersonalTranner());

ps\_perGYMInfo.setString(9, vo2.getFacility());

ps\_perGYMInfo.setLong(10, vo2.getRegistrationID());

if (ps\_perGYMInfo.executeUpdate() > 0 && ps\_perInfo.executeUpdate() > 0) {

con.commit();

return true;

}

} catch (MySQLIntegrityConstraintViolationException e) {

try {

con.rollback();

JOptionPane.showMessageDialog(FitnessDesk.getView(), "This record is already saved..");

} catch (SQLException e1) {

System.out.println("Error in Rollback new member joing dao");

}

e.printStackTrace();

} catch (Exception e) {

try {

con.rollback();

} catch (SQLException e1) {

System.out.println("Error in Rollback new member joing dao!");

}

e.printStackTrace();

}

return false;

}

}

**JoinginNewMemberView.java**

package com.cure.fit.member.view;

import java.awt.Color;

import java.awt.Font;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.DefaultComboBoxModel;

import javax.swing.ImageIcon;

import javax.swing.JButton;

import javax.swing.JComboBox;

import javax.swing.JInternalFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JTextArea;

import javax.swing.JTextField;

import javax.swing.SwingConstants;

import javax.swing.UIManager;

import javax.swing.border.TitledBorder;

import com.cure.fit.main.FitnessDesk;

import com.cure.fit.member.dao.DBOperation;

import com.cure.fit.member.dao.JoiningNewMemberDAOImpl;

import com.cure.fit.members.model.MemberGymInfoVO;

import com.cure.fit.members.model.MemberPersonalInfoVO;

import com.cure.fit.util.LoadImageUtil;

public class JoinginNewMemberView extends JInternalFrame {

/\*\*

\* serialVersionUID

\*/

private static final long serialVersionUID = 475059115660230894L;

private JTextField textField;

private JTextField textField\_1;

private JTextField textField\_2;

private JTextField textField\_3;

private JTextField textField\_4;

private JTextField textField\_6;

private JTextField textField\_7;

private JTextField textField\_8;

private JLabel lblLastName;

private JLabel lblDateOfBirth;

private JLabel lblContactNo;

private JLabel lblAllternateContact;

private JLabel lblGender;

private JLabel lblEmailAddress;

private JLabel lblOccupation;

private JLabel lblAddress;

private JLabel lblNewLabel\_1;

private JTextField textField\_10;

private JTextField textField\_12;

private JTextField textField\_13;

private JTextField textField\_14;

private DBOperation doa;

private MemberPersonalInfoVO perInfoVO = new MemberPersonalInfoVO();

private MemberGymInfoVO perGymInfo = new MemberGymInfoVO();

private JComboBox comboBox\_3;

private JComboBox comboBox\_2;

private JTextArea textArea;

private JComboBox comboBox\_1;

private JComboBox comboBox;

private JComboBox comboBox\_4;

/\*\*

\* Create the frame.

\*/

public JoinginNewMemberView() {

doa = JoiningNewMemberDAOImpl.getInstance();

setTitle("New Member Joinging");

getContentPane().setBackground(new Color(0, 255, 255));

setClosable(true);

setIconifiable(true);

// setMaximizable(true);

// setResizable(true);

setBounds(100, 100, 913, 536);

getContentPane().setLayout(null);

JPanel panel = new JPanel();

panel.setBackground(UIManager.getColor("InternalFrame.activeTitleGradient"));

panel.setBorder(new TitledBorder(null, "Personal Information", TitledBorder.LEADING, TitledBorder.TOP, null,

Color.GRAY));

panel.setBounds(187, 11, 700, 244);

getContentPane().add(panel);

panel.setLayout(null);

textField = new JTextField();

textField.setBounds(145, 26, 169, 31);

panel.add(textField);

textField.setColumns(10);

textField\_1 = new JTextField();

textField\_1.setBounds(145, 68, 169, 31);

panel.add(textField\_1);

textField\_1.setColumns(10);

textField\_2 = new JTextField();

textField\_2.setBounds(145, 110, 169, 31);

panel.add(textField\_2);

textField\_2.setColumns(10);

textField\_3 = new JTextField();

textField\_3.setBounds(145, 152, 169, 31);

panel.add(textField\_3);

textField\_3.setColumns(10);

textField\_4 = new JTextField();

textField\_4.setBounds(145, 194, 169, 31);

panel.add(textField\_4);

textField\_4.setColumns(10);

textField\_6 = new JTextField();

textField\_6.setBounds(487, 68, 176, 31);

panel.add(textField\_6);

textField\_6.setColumns(10);

textField\_7 = new JTextField();

textField\_7.setBounds(487, 110, 176, 31);

panel.add(textField\_7);

textField\_7.setColumns(10);

textArea = new JTextArea();

textArea.setBounds(487, 152, 176, 65);

panel.add(textArea);

JLabel lblNewLabel = new JLabel("First Name :");

lblNewLabel.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblNewLabel.setBounds(20, 29, 115, 28);

panel.add(lblNewLabel);

lblLastName = new JLabel("Last Name :");

lblLastName.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblLastName.setBounds(20, 68, 115, 28);

panel.add(lblLastName);

lblDateOfBirth = new JLabel("Date of Birth :");

lblDateOfBirth.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblDateOfBirth.setBounds(20, 110, 115, 28);

panel.add(lblDateOfBirth);

lblContactNo = new JLabel("Contact No :");

lblContactNo.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblContactNo.setBounds(20, 152, 115, 28);

panel.add(lblContactNo);

lblAllternateContact = new JLabel("Allternate Contact :");

lblAllternateContact.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblAllternateContact.setBounds(20, 194, 115, 28);

panel.add(lblAllternateContact);

lblGender = new JLabel("Gender :");

lblGender.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblGender.setBounds(362, 26, 115, 28);

panel.add(lblGender);

lblEmailAddress = new JLabel("Email Address :");

lblEmailAddress.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblEmailAddress.setBounds(362, 65, 115, 28);

panel.add(lblEmailAddress);

lblOccupation = new JLabel("Occupation :");

lblOccupation.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblOccupation.setBounds(362, 110, 115, 28);

panel.add(lblOccupation);

lblAddress = new JLabel("Address :");

lblAddress.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblAddress.setBounds(362, 152, 115, 28);

panel.add(lblAddress);

comboBox\_3 = new JComboBox();

comboBox\_3.setModel(new DefaultComboBoxModel(new String[] { "Male", "Female", "Others" }));

comboBox\_3.setBounds(487, 26, 176, 31);

panel.add(comboBox\_3);

JPanel panel\_1 = new JPanel();

panel\_1.setBackground(UIManager.getColor("InternalFrame.activeTitleGradient"));

panel\_1.setBorder(

new TitledBorder(null, "GYM Information", TitledBorder.LEADING, TitledBorder.TOP, null, Color.GRAY));

panel\_1.setBounds(187, 266, 700, 233);

getContentPane().add(panel\_1);

panel\_1.setLayout(null);

textField\_10 = new JTextField();

textField\_10.setColumns(10);

textField\_10.setBounds(141, 81, 176, 31);

panel\_1.add(textField\_10);

textField\_12 = new JTextField();

textField\_12.setColumns(10);

textField\_12.setBounds(141, 188, 176, 31);

panel\_1.add(textField\_12);

textField\_13 = new JTextField();

textField\_13.setColumns(10);

textField\_13.setBounds(488, 27, 176, 31);

panel\_1.add(textField\_13);

textField\_14 = new JTextField();

textField\_14.setColumns(10);

textField\_14.setBounds(488, 81, 176, 31);

panel\_1.add(textField\_14);

comboBox = new JComboBox();

comboBox.setBounds(488, 131, 176, 31);

comboBox.setModel(new DefaultComboBoxModel(((JoiningNewMemberDAOImpl) doa).getTrainnerName()));

panel\_1.add(comboBox);

comboBox\_1 = new JComboBox();

comboBox\_1.setModel(new DefaultComboBoxModel(

new String[] { "GYM", "GYM/Stream/Cardio", "GYM/Stream", "GYM/Cardio", "Aerobics/Power Yoga" }));

comboBox\_1.setBounds(488, 188, 176, 31);

panel\_1.add(comboBox\_1);

JLabel lblSubscription = new JLabel("Subscription :");

lblSubscription.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblSubscription.setBounds(16, 32, 115, 28);

panel\_1.add(lblSubscription);

comboBox\_2 = new JComboBox();

comboBox\_2.setModel(new DefaultComboBoxModel(new String[] { "Yearly", "Half Yearly", "Quaterly", "Monthly" }));

comboBox\_2.setBounds(141, 32, 176, 31);

panel\_1.add(comboBox\_2);

JLabel lblAmount = new JLabel("Amount :");

lblAmount.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblAmount.setBounds(16, 81, 115, 28);

panel\_1.add(lblAmount);

JLabel lblAmountStatus = new JLabel("Amount Status :");

lblAmountStatus.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblAmountStatus.setBounds(16, 131, 115, 28);

panel\_1.add(lblAmountStatus);

JLabel lblPendingAmount = new JLabel("Pending Amount :");

lblPendingAmount.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblPendingAmount.setBounds(16, 188, 115, 28);

panel\_1.add(lblPendingAmount);

JLabel lblStartDate = new JLabel("Start Date :");

lblStartDate.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblStartDate.setBounds(363, 27, 115, 28);

panel\_1.add(lblStartDate);

JLabel lblEndDate = new JLabel("End Date :");

lblEndDate.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblEndDate.setBounds(363, 81, 115, 28);

panel\_1.add(lblEndDate);

JLabel lblPersonalTrainner = new JLabel("Personal Trainner");

lblPersonalTrainner.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblPersonalTrainner.setBounds(363, 131, 115, 28);

panel\_1.add(lblPersonalTrainner);

JLabel lblFacility = new JLabel("Facility :");

lblFacility.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblFacility.setBounds(363, 188, 115, 28);

panel\_1.add(lblFacility);

comboBox\_4 = new JComboBox();

comboBox\_4.setModel(new DefaultComboBoxModel(new String[] { "Due Balance", "Paid" }));

comboBox\_4.setBounds(141, 136, 176, 31);

panel\_1.add(comboBox\_4);

JPanel panel\_2 = new JPanel();

panel\_2.setBackground(UIManager.getColor("InternalFrame.activeTitleGradient"));

panel\_2.setBounds(10, 11, 171, 488);

getContentPane().add(panel\_2);

panel\_2.setLayout(null);

// Import ImageIcon

// ImageIcon iconLogo = new

// ImageIcon("./images/com/cure/fit/profile/defaultUserPic.jpg");

lblNewLabel\_1 = new JLabel("Upload Picture");

lblNewLabel\_1.setHorizontalAlignment(SwingConstants.CENTER);

lblNewLabel\_1.setBackground(Color.GRAY);

lblNewLabel\_1.setOpaque(true);

lblNewLabel\_1.setBounds(10, 11, 151, 145);

String defualtImgPath = "./images/com/cure/fit/profile/defaultUserPic.jpg";

lblNewLabel\_1.setIcon(LoadImageUtil.resizeImage(lblNewLabel\_1, defualtImgPath));

perInfoVO.setImagePath(defualtImgPath);

panel\_2.add(lblNewLabel\_1);

JButton btnNewButton = new JButton("Upload");

btnNewButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

String imgPath = LoadImageUtil.loadImage(lblNewLabel\_1);

perInfoVO.setImagePath(imgPath);

}

});

btnNewButton.setBounds(10, 167, 151, 31);

panel\_2.add(btnNewButton);

JLabel lblNewLabel\_2 = new JLabel("Registration ID");

lblNewLabel\_2.setBounds(10, 221, 151, 31);

panel\_2.add(lblNewLabel\_2);

textField\_8 = new JTextField();

textField\_8.setEditable(false);

textField\_8.setText(((JoiningNewMemberDAOImpl) doa).getNextGeneratedID() + "");

textField\_8.setBounds(10, 253, 74, 31);

panel\_2.add(textField\_8);

textField\_8.setColumns(10);

JButton btnNewButton\_1 = new JButton("PRINT BILL");

btnNewButton\_1.setFont(new Font("Tahoma", Font.BOLD, 11));

btnNewButton\_1.setBounds(10, 377, 151, 31);

panel\_2.add(btnNewButton\_1);

JButton button = new JButton("SAVE");

button.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if (loadDataToPersonalInfoVO()) {

boolean saveStatus = doa.save(perInfoVO, perGymInfo);

if (saveStatus)

JOptionPane.showMessageDialog(FitnessDesk.getView(), "The record is saved successfuly");

else

JOptionPane.showMessageDialog(FitnessDesk.getView(), "The record is not saved successfuly");

} else {

JOptionPane.showMessageDialog(FitnessDesk.getView(), "Please fill all the fields propertly");

}

}

});

button.setFont(new Font("Tahoma", Font.BOLD, 11));

button.setBounds(10, 430, 151, 31);

panel\_2.add(button);

}

public boolean loadDataToPersonalInfoVO() {

boolean perInfoFlag = false;

boolean perGymFlag = false;

if (!textField\_8.getText().equals("") && !textField.getText().equals("") && !textField\_1.getText().equals("")

&& !textField\_2.getText().equals("") && !textField\_3.getText().equals("")

&& !textField\_4.getText().equals("") && comboBox\_3.getSelectedItem() != null

&& !textField\_6.getText().equals("") && !textField\_7.getText().equals("")

&& !textArea.getText().equals("")) {

perInfoVO.setRegistrationID(Integer.parseInt(textField\_8.getText()));

perInfoVO.setFirstName(textField.getText());

perInfoVO.setLastName(textField\_1.getText());

perInfoVO.setDob(textField\_2.getText());

perInfoVO.setPrimaryNumber(Long.parseLong(textField\_3.getText()));

perInfoVO.setAlternetNumber(Long.parseLong(textField\_4.getText()));

perInfoVO.setGender(comboBox\_3.getSelectedItem().toString());

perInfoVO.setOccupation(textField\_6.getText());

perInfoVO.setEmail(textField\_7.getText());

perInfoVO.setAddress(textArea.getText());

perInfoFlag = true;

}

if (!textField\_8.getText().equals("") && !textField\_10.getText().equals("")

&& !textField\_12.getText().equals("") && !textField\_13.getText().equals("")

&& !textField\_14.getText().equals("")) {

perGymInfo.setRegistrationID(Long.valueOf(textField\_8.getText()));

perGymInfo.setSubscription(comboBox\_2.getSelectedItem().toString());

perGymInfo.setPaidAmount(Double.valueOf(textField\_10.getText()));

perGymInfo.setStatus(comboBox\_4.getSelectedItem().toString());

perGymInfo.setPaidAmount(Double.valueOf(textField\_12.getText()));

perGymInfo.setStartDate(textField\_13.getText());

perGymInfo.setEndDate(textField\_14.getText());

perGymInfo.setPersonalTranner(comboBox.getSelectedItem().toString());

perGymInfo.setFacility(comboBox\_1.getSelectedItem().toString());

perGymFlag = true;

}

if (perGymFlag && perInfoFlag)

return true;

else

return false;

}

}

**UpdateMemberDetailsView.java**

package com.cure.fit.member.view;

import java.awt.Color;

import java.awt.Font;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.util.List;

import javax.swing.DefaultComboBoxModel;

import javax.swing.JButton;

import javax.swing.JComboBox;

import javax.swing.JInternalFrame;

import javax.swing.JLabel;

import javax.swing.JOptionPane;

import javax.swing.JPanel;

import javax.swing.JTextArea;

import javax.swing.JTextField;

import javax.swing.SwingConstants;

import javax.swing.UIManager;

import javax.swing.border.TitledBorder;

import com.cure.fit.main.FitnessDesk;

import com.cure.fit.member.dao.DBOperation;

import com.cure.fit.member.dao.JoiningNewMemberDAOImpl;

import com.cure.fit.members.model.MemberGymInfoVO;

import com.cure.fit.members.model.MemberInfo;

import com.cure.fit.members.model.MemberPersonalInfoVO;

import com.cure.fit.util.LoadImageUtil;

public class UpdateMemberDetailsView extends JInternalFrame {

/\*\*

\* serialVersionUID

\*/

private static final long serialVersionUID = 475059115660230894L;

private DBOperation doa;

private MemberPersonalInfoVO perInfoVO = new MemberPersonalInfoVO();

private MemberGymInfoVO perGymInfo = new MemberGymInfoVO();

private JTextField textField;

private JTextField textField\_1;

private JTextField textField\_2;

private JTextField textField\_3;

private JTextField textField\_4;

private JTextField textField\_6;

private JTextField textField\_7;

private JTextField textField\_8;

private JLabel lblLastName;

private JLabel lblDateOfBirth;

private JLabel lblContactNo;

private JLabel lblAllternateContact;

private JLabel lblGender;

private JLabel lblEmailAddress;

private JLabel lblOccupation;

private JLabel lblAddress;

private JLabel lblNewLabel\_1;

private JTextField textField\_10;

private JTextField textField\_12;

private JTextField textField\_13;

private JTextField textField\_14;

private JComboBox comboBox\_3;

private JComboBox comboBox\_2;

private JTextArea textArea;

private JComboBox comboBox\_1;

private JComboBox comboBox;

private JComboBox comboBox\_4;

/\*\*

\* Create the frame.

\*/

public UpdateMemberDetailsView() {

doa = JoiningNewMemberDAOImpl.getInstance();

setTitle("Update Member Joinging");

getContentPane().setBackground(new Color(0, 255, 255));

setClosable(true);

setIconifiable(true);

// setMaximizable(true);

// setResizable(true);

setBounds(100, 100, 913, 536);

getContentPane().setLayout(null);

JPanel panel = new JPanel();

panel.setBackground(UIManager.getColor("InternalFrame.activeTitleGradient"));

panel.setBorder(new TitledBorder(null, "Personal Information", TitledBorder.LEADING, TitledBorder.TOP, null,

Color.GRAY));

panel.setBounds(187, 11, 700, 244);

getContentPane().add(panel);

panel.setLayout(null);

textField = new JTextField();

textField.setBounds(145, 26, 169, 31);

panel.add(textField);

textField.setColumns(10);

textField\_1 = new JTextField();

textField\_1.setBounds(145, 68, 169, 31);

panel.add(textField\_1);

textField\_1.setColumns(10);

textField\_2 = new JTextField();

textField\_2.setBounds(145, 110, 169, 31);

panel.add(textField\_2);

textField\_2.setColumns(10);

textField\_3 = new JTextField();

textField\_3.setBounds(145, 152, 169, 31);

panel.add(textField\_3);

textField\_3.setColumns(10);

textField\_4 = new JTextField();

textField\_4.setBounds(145, 194, 169, 31);

panel.add(textField\_4);

textField\_4.setColumns(10);

textField\_6 = new JTextField();

textField\_6.setBounds(487, 68, 176, 31);

panel.add(textField\_6);

textField\_6.setColumns(10);

textField\_7 = new JTextField();

textField\_7.setBounds(487, 110, 176, 31);

panel.add(textField\_7);

textField\_7.setColumns(10);

textArea = new JTextArea();

textArea.setBounds(487, 152, 176, 65);

panel.add(textArea);

JLabel lblNewLabel = new JLabel("First Name :");

lblNewLabel.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblNewLabel.setBounds(20, 29, 115, 28);

panel.add(lblNewLabel);

lblLastName = new JLabel("Last Name :");

lblLastName.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblLastName.setBounds(20, 68, 115, 28);

panel.add(lblLastName);

lblDateOfBirth = new JLabel("Date of Birth :");

lblDateOfBirth.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblDateOfBirth.setBounds(20, 110, 115, 28);

panel.add(lblDateOfBirth);

lblContactNo = new JLabel("Contact No :");

lblContactNo.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblContactNo.setBounds(20, 152, 115, 28);

panel.add(lblContactNo);

lblAllternateContact = new JLabel("Allternate Contact :");

lblAllternateContact.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblAllternateContact.setBounds(20, 194, 115, 28);

panel.add(lblAllternateContact);

lblGender = new JLabel("Gender :");

lblGender.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblGender.setBounds(362, 26, 115, 28);

panel.add(lblGender);

lblEmailAddress = new JLabel("Email Address :");

lblEmailAddress.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblEmailAddress.setBounds(362, 65, 115, 28);

panel.add(lblEmailAddress);

lblOccupation = new JLabel("Occupation :");

lblOccupation.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblOccupation.setBounds(362, 110, 115, 28);

panel.add(lblOccupation);

lblAddress = new JLabel("Address :");

lblAddress.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblAddress.setBounds(362, 152, 115, 28);

panel.add(lblAddress);

comboBox\_3 = new JComboBox();

comboBox\_3.setModel(new DefaultComboBoxModel(new String[] { "Male", "Female", "Others" }));

comboBox\_3.setBounds(487, 26, 176, 31);

panel.add(comboBox\_3);

JPanel panel\_1 = new JPanel();

panel\_1.setBackground(UIManager.getColor("InternalFrame.activeTitleGradient"));

panel\_1.setBorder(

new TitledBorder(null, "GYM Information", TitledBorder.LEADING, TitledBorder.TOP, null, Color.GRAY));

panel\_1.setBounds(187, 266, 700, 233);

getContentPane().add(panel\_1);

panel\_1.setLayout(null);

textField\_10 = new JTextField();

textField\_10.setColumns(10);

textField\_10.setBounds(141, 81, 176, 31);

panel\_1.add(textField\_10);

textField\_12 = new JTextField();

textField\_12.setColumns(10);

textField\_12.setBounds(141, 188, 176, 31);

panel\_1.add(textField\_12);

textField\_13 = new JTextField();

textField\_13.setColumns(10);

textField\_13.setBounds(488, 27, 176, 31);

panel\_1.add(textField\_13);

textField\_14 = new JTextField();

textField\_14.setColumns(10);

textField\_14.setBounds(488, 81, 176, 31);

panel\_1.add(textField\_14);

comboBox = new JComboBox();

comboBox.setBounds(488, 131, 176, 31);

comboBox.setModel(new DefaultComboBoxModel(((JoiningNewMemberDAOImpl) doa).getTrainnerName()));

panel\_1.add(comboBox);

comboBox\_1 = new JComboBox();

comboBox\_1.setModel(new DefaultComboBoxModel(

new String[] { "GYM", "GYM/Stream/Cardio", "GYM/Stream", "GYM/Cardio", "Aerobics/Power Yoga" }));

comboBox\_1.setBounds(488, 188, 176, 31);

panel\_1.add(comboBox\_1);

JLabel lblSubscription = new JLabel("Subscription :");

lblSubscription.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblSubscription.setBounds(16, 32, 115, 28);

panel\_1.add(lblSubscription);

comboBox\_2 = new JComboBox();

comboBox\_2.setModel(new DefaultComboBoxModel(new String[] { "Yearly", "Half Yearly", "Quaterly", "Monthly" }));

comboBox\_2.setBounds(141, 32, 176, 31);

panel\_1.add(comboBox\_2);

JLabel lblAmount = new JLabel("Amount :");

lblAmount.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblAmount.setBounds(16, 81, 115, 28);

panel\_1.add(lblAmount);

JLabel lblAmountStatus = new JLabel("Amount Status :");

lblAmountStatus.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblAmountStatus.setBounds(16, 131, 115, 28);

panel\_1.add(lblAmountStatus);

JLabel lblPendingAmount = new JLabel("Pending Amount :");

lblPendingAmount.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblPendingAmount.setBounds(16, 188, 115, 28);

panel\_1.add(lblPendingAmount);

JLabel lblStartDate = new JLabel("Start Date :");

lblStartDate.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblStartDate.setBounds(363, 27, 115, 28);

panel\_1.add(lblStartDate);

JLabel lblEndDate = new JLabel("End Date :");

lblEndDate.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblEndDate.setBounds(363, 81, 115, 28);

panel\_1.add(lblEndDate);

JLabel lblPersonalTrainner = new JLabel("Personal Trainner");

lblPersonalTrainner.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblPersonalTrainner.setBounds(363, 131, 115, 28);

panel\_1.add(lblPersonalTrainner);

JLabel lblFacility = new JLabel("Facility :");

lblFacility.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblFacility.setBounds(363, 188, 115, 28);

panel\_1.add(lblFacility);

comboBox\_4 = new JComboBox();

comboBox\_4.setModel(new DefaultComboBoxModel(new String[] { "Due Balance", "Paid" }));

comboBox\_4.setBounds(141, 136, 176, 31);

panel\_1.add(comboBox\_4);

JPanel panel\_2 = new JPanel();

panel\_2.setBackground(UIManager.getColor("InternalFrame.activeTitleGradient"));

panel\_2.setBounds(10, 11, 171, 488);

getContentPane().add(panel\_2);

panel\_2.setLayout(null);

// Import ImageIcon

// ImageIcon iconLogo = new

// ImageIcon("./images/com/cure/fit/profile/defaultUserPic.jpg");

lblNewLabel\_1 = new JLabel("Upload Picture");

lblNewLabel\_1.setHorizontalAlignment(SwingConstants.CENTER);

lblNewLabel\_1.setBackground(Color.GRAY);

lblNewLabel\_1.setOpaque(true);

lblNewLabel\_1.setBounds(10, 11, 151, 145);

String defualtImgPath = "./images/com/cure/fit/profile/defaultUserPic.jpg";

lblNewLabel\_1.setIcon(LoadImageUtil.resizeImage(lblNewLabel\_1, defualtImgPath));

perInfoVO.setImagePath(defualtImgPath);

panel\_2.add(lblNewLabel\_1);

JButton btnNewButton = new JButton("Upload");

btnNewButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

String imgPath = LoadImageUtil.loadImage(lblNewLabel\_1);

perInfoVO.setImagePath(imgPath);

}

});

btnNewButton.setBounds(10, 167, 151, 31);

panel\_2.add(btnNewButton);

JButton btnNewButton\_1 = new JButton("PRINT BILL");

btnNewButton\_1.setFont(new Font("Tahoma", Font.BOLD, 11));

btnNewButton\_1.setBounds(10, 377, 151, 31);

panel\_2.add(btnNewButton\_1);

JButton btnUpdate = new JButton("UPDATE");

btnUpdate.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if (loadDataToPersonalInfoVO()) {

boolean saveStatus = doa.update(perInfoVO, perGymInfo);

if (saveStatus)

JOptionPane.showMessageDialog(FitnessDesk.getView(), "The record is updated successfuly");

else

JOptionPane.showMessageDialog(FitnessDesk.getView(), "The record is fail to update");

} else {

JOptionPane.showMessageDialog(FitnessDesk.getView(), "Please fill all the fields propertly");

}

}

});

btnUpdate.setFont(new Font("Tahoma", Font.BOLD, 11));

btnUpdate.setBounds(10, 430, 151, 31);

panel\_2.add(btnUpdate);

JPanel panel\_3 = new JPanel();

panel\_3.setOpaque(false);

panel\_3.setBorder(new TitledBorder(null, "Search By ID", TitledBorder.LEADING, TitledBorder.TOP, null, null));

panel\_3.setBounds(10, 209, 151, 132);

panel\_2.add(panel\_3);

panel\_3.setLayout(null);

JLabel lblNewLabel\_2 = new JLabel("Registration ID :");

lblNewLabel\_2.setBounds(21, 22, 108, 14);

panel\_3.add(lblNewLabel\_2);

textField\_8 = new JTextField();

textField\_8.setBounds(21, 47, 108, 31);

panel\_3.add(textField\_8);

//textField\_8.setText(((JoiningNewMemberDAOImpl) doa).getNextGeneratedID() + "");

textField\_8.setColumns(10);

JButton btnSearch = new JButton("SEARCH");

btnSearch.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

List<MemberInfo> listVO = ((JoiningNewMemberDAOImpl) doa)

.searchMemberByID(Long.valueOf(textField\_8.getText()));

for (MemberInfo memberInfo : listVO) {

if (memberInfo instanceof MemberGymInfoVO) {

perGymInfo = (MemberGymInfoVO) memberInfo;

} else {

perInfoVO = (MemberPersonalInfoVO) memberInfo;

}

}

updateMemberUI();

}

});

btnSearch.setFont(new Font("Tahoma", Font.BOLD, 11));

btnSearch.setBounds(21, 90, 108, 31);

panel\_3.add(btnSearch);

}

public boolean loadDataToPersonalInfoVO() {

boolean perInfoFlag = false;

boolean perGymFlag = false;

if (!textField\_8.getText().equals("") && !textField.getText().equals("") && !textField\_1.getText().equals("")

&& !textField\_2.getText().equals("") && !textField\_3.getText().equals("")

&& !textField\_4.getText().equals("") && comboBox\_3.getSelectedItem() != null

&& !textField\_6.getText().equals("") && !textField\_7.getText().equals("")

&& !textArea.getText().equals("")) {

perInfoVO.setRegistrationID(Integer.parseInt(textField\_8.getText()));

perInfoVO.setFirstName(textField.getText());

perInfoVO.setLastName(textField\_1.getText());

perInfoVO.setDob(textField\_2.getText());

perInfoVO.setPrimaryNumber(Long.parseLong(textField\_3.getText()));

perInfoVO.setAlternetNumber(Long.parseLong(textField\_4.getText()));

perInfoVO.setGender(comboBox\_3.getSelectedItem().toString());

perInfoVO.setOccupation(textField\_6.getText());

perInfoVO.setEmail(textField\_7.getText());

perInfoVO.setAddress(textArea.getText());

perInfoFlag = true;

}

if (!textField\_8.getText().equals("") && !textField\_10.getText().equals("")

&& !textField\_12.getText().equals("") && !textField\_13.getText().equals("")

&& !textField\_14.getText().equals("")) {

perGymInfo.setRegistrationID(Long.valueOf(textField\_8.getText()));

perGymInfo.setSubscription(comboBox\_2.getSelectedItem().toString());

perGymInfo.setPaidAmount(Double.valueOf(textField\_10.getText()));

perGymInfo.setStatus(comboBox\_4.getSelectedItem().toString());

perGymInfo.setPaidAmount(Double.valueOf(textField\_12.getText()));

perGymInfo.setStartDate(textField\_13.getText());

perGymInfo.setEndDate(textField\_14.getText());

perGymInfo.setPersonalTranner(comboBox.getSelectedItem().toString());

perGymInfo.setFacility(comboBox\_1.getSelectedItem().toString());

perGymFlag = true;

}

if (perGymFlag && perInfoFlag)

return true;

else

return false;

}

public void updateMemberUI() {

if (perInfoVO != null) {

textField\_8.setText("" + perInfoVO.getRegistrationID());

textField.setText(perInfoVO.getFirstName());

textField\_1.setText(perInfoVO.getLastName());

textField\_2.setText(perInfoVO.getDob());

textField\_3.setText("" + perInfoVO.getPrimaryNumber());

textField\_4.setText("" + perInfoVO.getAlternetNumber());

comboBox\_3.setSelectedItem(perInfoVO.getGender());

textField\_6.setText(perInfoVO.getOccupation());

textField\_7.setText(perInfoVO.getEmail());

textArea.setText(perInfoVO.getAddress());

}

if (perGymInfo != null) {

textField\_8.setText("" + perGymInfo.getRegistrationID());

comboBox\_2.setSelectedItem(perGymInfo.getSubscription());

textField\_10.setText("" + perGymInfo.getPaidAmount());

comboBox\_4.setSelectedItem(perGymInfo.getStatus());

textField\_12.setText("" + perGymInfo.getPendingAmount());

textField\_13.setText(perGymInfo.getStartDate());

textField\_14.setText(perGymInfo.getEndDate());

comboBox.setSelectedItem(perGymInfo.getPersonalTranner());

comboBox\_1.setSelectedItem(perGymInfo.getFacility());

}

}

}

MemberGymInfoVO.java

package com.cure.fit.members.model;

public class MemberGymInfoVO extends MemberInfo{

private long registrationID;

private String subscription;

private double paidAmount;

private double pendingAmount;

private String status;

private String startDate;

private String endDate;

private String personalTranner;

private String facility;

/\*\*

\* @return the subscription

\*/

public String getSubscription() {

return subscription;

}

/\*\*

\* @param subscription the subscription to set

\*/

public void setSubscription(String subscription) {

this.subscription = subscription;

}

/\*\*

\* @return the paidAmount

\*/

public double getPaidAmount() {

return paidAmount;

}

/\*\*

\* @param paidAmount the paidAmount to set

\*/

public void setPaidAmount(double paidAmount) {

this.paidAmount = paidAmount;

}

/\*\*

\* @return the pendingAmount

\*/

public double getPendingAmount() {

return pendingAmount;

}

/\*\*

\* @param pendingAmount the pendingAmount to set

\*/

public void setPendingAmount(double pendingAmount) {

this.pendingAmount = pendingAmount;

}

/\*\*

\* @return the status

\*/

public String getStatus() {

return status;

}

/\*\*

\* @param status the status to set

\*/

public void setStatus(String status) {

this.status = status;

}

/\*\*

\* @return the startDate

\*/

public String getStartDate() {

return startDate;

}

/\*\*

\* @param startDate the startDate to set

\*/

public void setStartDate(String startDate) {

this.startDate = startDate;

}

/\*\*

\* @return the endDate

\*/

public String getEndDate() {

return endDate;

}

/\*\*

\* @param endDate the endDate to set

\*/

public void setEndDate(String endDate) {

this.endDate = endDate;

}

/\*\*

\* @return the personalTranner

\*/

public String getPersonalTranner() {

return personalTranner;

}

/\*\*

\* @param personalTranner the personalTranner to set

\*/

public void setPersonalTranner(String personalTranner) {

this.personalTranner = personalTranner;

}

/\*\*

\* @return the facility

\*/

public String getFacility() {

return facility;

}

/\*\*

\* @param facility the facility to set

\*/

public void setFacility(String facility) {

this.facility = facility;

}

}

MemberInfo.java

package com.cure.fit.members.model;

public class MemberInfo {

long registrationID;

public long getRegistrationID() {

return registrationID;

}

public void setRegistrationID(long registrationID) {

this.registrationID = registrationID;

}

}

MemberPersonalInfoVO.java

package com.cure.fit.members.model;

public class MemberPersonalInfoVO extends MemberInfo{

private String firstName;

private String lastName;

private String gender;

private String dob;

private long primaryNumber;

private long alternetNumber;

private String email;

private String occupation;

private String address;

private String imagePath;

/\*\*

\* @return the firstName

\*/

public String getFirstName() {

return firstName;

}

/\*\*

\* @param firstName the firstName to set

\*/

public void setFirstName(String firstName) {

this.firstName = firstName;

}

/\*\*

\* @return the lastName

\*/

public String getLastName() {

return lastName;

}

/\*\*

\* @param lastName the lastName to set

\*/

public void setLastName(String lastName) {

this.lastName = lastName;

}

/\*\*

\* @return the gender

\*/

public String getGender() {

return gender;

}

/\*\*

\* @param gender the gender to set

\*/

public void setGender(String gender) {

this.gender = gender;

}

/\*\*

\* @return the dob

\*/

public String getDob() {

return dob;

}

/\*\*

\* @param dob the dob to set

\*/

public void setDob(String dob) {

this.dob = dob;

}

/\*\*

\* @return the primaryNumber

\*/

public long getPrimaryNumber() {

return primaryNumber;

}

/\*\*

\* @param primaryNumber the primaryNumber to set

\*/

public void setPrimaryNumber(long primaryNumber) {

this.primaryNumber = primaryNumber;

}

/\*\*

\* @return the alternetNumber

\*/

public long getAlternetNumber() {

return alternetNumber;

}

/\*\*

\* @param alternetNumber the alternetNumber to set

\*/

public void setAlternetNumber(long alternetNumber) {

this.alternetNumber = alternetNumber;

}

/\*\*

\* @return the email

\*/

public String getEmail() {

return email;

}

/\*\*

\* @param email the email to set

\*/

public void setEmail(String email) {

this.email = email;

}

/\*\*

\* @return the occupation

\*/

public String getOccupation() {

return occupation;

}

/\*\*

\* @param occupation the occupation to set

\*/

public void setOccupation(String occupation) {

this.occupation = occupation;

}

/\*\*

\* @return the address

\*/

public String getAddress() {

return address;

}

/\*\*

\* @param address the address to set

\*/

public void setAddress(String address) {

this.address = address;

}

public String getImagePath() {

return imagePath;

}

public void setImagePath(String imagePath) {

this.imagePath = imagePath;

}

}

**BalanceFeesView.java**

package com.cure.fit.transaction.view;

import java.awt.EventQueue;

import javax.swing.JInternalFrame;

import java.awt.Color;

import javax.swing.JPanel;

import javax.swing.border.SoftBevelBorder;

import javax.swing.border.BevelBorder;

import javax.swing.JLabel;

import java.awt.Font;

import javax.swing.JComboBox;

import javax.swing.JButton;

import javax.swing.DefaultComboBoxModel;

import javax.swing.JScrollPane;

import javax.swing.JTable;

import javax.swing.table.DefaultTableModel;

public class BalanceFeesView extends JInternalFrame {

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

private JTable table;

/\*\*

\* Launch the application.

\*/

public static void main(String[] args) {

EventQueue.invokeLater(new Runnable() {

public void run() {

try {

BalanceFeesView frame = new BalanceFeesView();

frame.setVisible(true);

} catch (Exception e) {

e.printStackTrace();

}

}

});

}

/\*\*

\* Create the frame.

\*/

public BalanceFeesView() {

getContentPane().setBackground(Color.CYAN);

setBounds(100, 100, 823, 462);

getContentPane().setLayout(null);

setTitle("Balance Fee Details");

setClosable(true);

setIconifiable(true);

JPanel panel = new JPanel();

panel.setOpaque(false);

panel.setBorder(new SoftBevelBorder(BevelBorder.LOWERED, null, null, null, null));

panel.setBounds(10, 11, 787, 95);

getContentPane().add(panel);

panel.setLayout(null);

JLabel lblFeeList = new JLabel("Fee List");

lblFeeList.setFont(new Font("Tahoma", Font.BOLD, 24));

lblFeeList.setForeground(Color.DARK\_GRAY);

lblFeeList.setBounds(307, 11, 119, 29);

panel.add(lblFeeList);

JLabel lblFeeStatus = new JLabel("Fee Status :");

lblFeeStatus.setFont(new Font("Tahoma", Font.PLAIN, 12));

lblFeeStatus.setBounds(217, 51, 94, 32);

panel.add(lblFeeStatus);

JComboBox comboBox = new JComboBox();

comboBox.setModel(new DefaultComboBoxModel(new String[] { "Paid", "Due " }));

comboBox.setBounds(307, 51, 158, 33);

panel.add(comboBox);

JButton btnNewButton = new JButton("Search");

btnNewButton.setFont(new Font("Tahoma", Font.BOLD, 11));

btnNewButton.setBounds(501, 51, 108, 32);

panel.add(btnNewButton);

JPanel panel\_1 = new JPanel();

panel\_1.setOpaque(false);

panel\_1.setBorder(new SoftBevelBorder(BevelBorder.LOWERED, null, null, null, null));

panel\_1.setBounds(10, 109, 787, 312);

getContentPane().add(panel\_1);

panel\_1.setLayout(null);

JScrollPane scrollPane = new JScrollPane();

scrollPane.setBounds(10, 11, 767, 290);

panel\_1.add(scrollPane);

table = new JTable();

table.setModel(new DefaultTableModel(new String[] { "RegID", "firstName", "lastName", "gender", "primaryNumber",

"email", "subscription", "paidAmount", "pendingAmount", "personalTranner", "facility" }, 0));

scrollPane.setViewportView(table);

}

}

**DateLabelFormatter.java**

package com.cure.fit.util;

import java.text.ParseException;

import java.text.SimpleDateFormat;

import java.util.Calendar;

import javax.swing.JFormattedTextField.AbstractFormatter;

public class DateLabelFormatter extends AbstractFormatter {

private String datePattern = "yyyy-MM-dd";

private SimpleDateFormat dateFormatter = new SimpleDateFormat(datePattern);

@Override

public Object stringToValue(String text) throws ParseException {

return dateFormatter.parseObject(text);

}

@Override

public String valueToString(Object value) throws ParseException {

if (value != null) {

Calendar cal = (Calendar) value;

return dateFormatter.format(cal.getTime());

}

return "";

}

}

**LoadImageUtil.java**

package com.cure.fit.util;

import java.awt.Image;

import java.io.File;

import javax.swing.ImageIcon;

import javax.swing.JFileChooser;

import javax.swing.JLabel;

import javax.swing.filechooser.FileNameExtensionFilter;

public class LoadImageUtil {

public static String loadImage(JLabel label) {

JFileChooser file = new JFileChooser();

file.setCurrentDirectory(new File(System.getProperty("user.home")));

// filter the files

FileNameExtensionFilter filter = new FileNameExtensionFilter("\*.Images", "jpg", "gif", "png");

file.addChoosableFileFilter(filter);

int result = file.showSaveDialog(null);

// if the user click on save in Jfilechooser

if (result == JFileChooser.APPROVE\_OPTION) {

File selectedFile = file.getSelectedFile();

String path = selectedFile.getAbsolutePath();

label.setIcon(resizeImage(label,path));

return path;

}

// if the user click on save in Jfilechooser

else if (result == JFileChooser.CANCEL\_OPTION) {

System.out.println("No File Select");

}

return "";

}

// Methode to resize imageIcon with the same size of a Jlabel

public static ImageIcon resizeImage(JLabel label,String ImagePath) {

ImageIcon MyImage = new ImageIcon(ImagePath);

Image img = MyImage.getImage();

Image newImg = img.getScaledInstance(label.getWidth(), label.getHeight(), Image.SCALE\_SMOOTH);

ImageIcon image = new ImageIcon(newImg);

return image;

}

}

**MysqlConnectionUtil.java**

package com.cure.fit.util;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

public class MysqlConnectionUtil {

private static Connection con;

public static Connection createConnection()

{

try {

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

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

} catch (ClassNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return con;

}

}