Joseph Griffith

12/12/2020

Final Project Code

DAO.JAVA

import java.io.BufferedReader;

import java.io.File;

import java.io.FileReader;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.Arrays;

import java.util.List;

import javaapplication1.Dao;

public class Dao {

static Connection connect = null;

Statement statement = null;

public Connection getConnection() {

try {

connect =

DriverManager.getConnection("jdbc:mysql://www.papademas.net:3307/tickets?autoReconnect=true&useSSL=false&user=fp411&password=411");

} catch (SQLException e) {

e.printStackTrace();

}

return connect;

}

public void createTables() {

String createTicketsTable = "CREATE TABLE jgriff\_tickets(ticket\_id INT AUTO\_INCREMENT PRIMARY KEY, ticket\_issuer VARCHAR(30), ticket\_description VARCHAR(200))";

String createUsersTable = "CREATE TABLE jgriff\_users(uid INT AUTO\_INCREMENT PRIMARY KEY, uname VARCHAR(30), upass VARCHAR(30), admin int)";

try {

this.statement = getConnection().createStatement();

this.statement.executeUpdate("CREATE TABLE jgriff\_tickets(ticket\_id INT AUTO\_INCREMENT PRIMARY KEY, ticket\_issuer VARCHAR(30), ticket\_description VARCHAR(200))");

this.statement.executeUpdate("CREATE TABLE jgriff\_users(uid INT AUTO\_INCREMENT PRIMARY KEY, uname VARCHAR(30), upass VARCHAR(30), admin int)");

System.out.println("Created tables in given database...");

this.statement.close();

connect.close();

} catch (Exception e) {

System.out.println(e.getMessage());

}

addUsers();

}

public void addUsers() {

List<List<String>> array = new ArrayList<>();

try {

BufferedReader br = new BufferedReader(new FileReader(new File("./userlist.csv")));

String line;

while ((line = br.readLine()) != null)

array.add(Arrays.asList(line.split(",")));

} catch (Exception e) {

String line;

System.out.println("There was a problem loading the file");

}

try {

Statement statement = getConnection().createStatement();

for (List<String> rowData : array) {

String sql = "insert into jgriff\_users(uname,upass,admin) values('" + (String)rowData.get(0) + "'," + " '" +

(String)rowData.get(1) + "','" + (String)rowData.get(2) + "');";

statement.executeUpdate(sql);

}

System.out.println("Inserts completed in the given database...");

statement.close();

} catch (Exception e) {

System.out.println(e.getMessage());

}

}

public int insertRecords(String ticketName, String ticketDesc) {

int id = 0;

try {

this.statement = getConnection().createStatement();

this.statement.executeUpdate("Insert into jgriff\_tickets(ticket\_issuer, ticket\_description) values( '" +

ticketName + "','" + ticketDesc + "')", 1);

ResultSet resultSet = null;

resultSet = this.statement.getGeneratedKeys();

if (resultSet.next())

id = resultSet.getInt(1);

} catch (SQLException e) {

e.printStackTrace();

}

return id;

}

public ResultSet readRecords() {

ResultSet results = null;

try {

this.statement = connect.createStatement();

results = this.statement.executeQuery("SELECT \* FROM jgriff\_tickets");

} catch (SQLException e1) {

e1.printStackTrace();

}

return results;

}

public ResultSet selectRecords(int ticketID) {

ResultSet results = null;

try {

this.statement = connect.createStatement();

results = this.statement.executeQuery("SELECT \* FROM jgriff\_tickets WHERE ticket\_id = " + ticketID);

connect.close();

} catch (SQLException e1) {

e1.printStackTrace();

}

return results;

}

}

LOGIN.JAVA

import java.awt.GridLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javaapplication1.Dao;

import javaapplication1.Login;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPasswordField;

import javax.swing.JTextField;

public class Login extends JFrame {

Dao conn;

public Login() {

super("IIT HELP DESK LOGIN");

this.conn = new Dao();

this.conn.createTables();

setSize(400, 210);

setLayout(new GridLayout(4, 2));

setLocationRelativeTo(null);

JLabel lblUsername = new JLabel("Username", 2);

JLabel lblPassword = new JLabel("Password", 2);

JLabel lblStatus = new JLabel(" ", 0);

JTextField txtUname = new JTextField(10);

JPasswordField txtPassword = new JPasswordField();

JButton btn = new JButton("Submit");

JButton btnExit = new JButton("Exit");

lblStatus.setToolTipText("Contact help desk to unlock password");

lblUsername.setHorizontalAlignment(0);

lblPassword.setHorizontalAlignment(0);

add(lblUsername);

add(txtUname);

add(lblPassword);

add(txtPassword);

add(btn);

add(btnExit);

add(lblStatus);

btn.addActionListener((ActionListener)new Object(this, txtUname, txtPassword, lblStatus));

btnExit.addActionListener(e -> System.exit(0));

setVisible(true);

}

public static void main(String[] args) {}

}

TICKETS.JAVA

import java.awt.Color;

import java.awt.Component;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.awt.event.WindowListener;

import java.sql.SQLException;

import javaapplication1.Dao;

import javaapplication1.Tickets;

import javaapplication1.ticketsJTable;

import javax.swing.JFrame;

import javax.swing.JMenu;

import javax.swing.JMenuBar;

import javax.swing.JMenuItem;

import javax.swing.JOptionPane;

import javax.swing.JScrollPane;

import javax.swing.JTable;

public class Tickets extends JFrame implements ActionListener {

Dao dao = new Dao();

Boolean chkIfAdmin = null;

private JMenu mnuFile = new JMenu("File");

private JMenu mnuAdmin = new JMenu("Admin");

private JMenu mnuTickets = new JMenu("Tickets");

JMenuItem mnuItemExit;

JMenuItem mnuItemUpdate;

JMenuItem mnuItemDelete;

JMenuItem mnuItemOpenTicket;

JMenuItem mnuItemViewTicket;

public Tickets(Boolean isAdmin) {

if (isAdmin != this.chkIfAdmin) {

System.out.println("You are an admin");

}

this.chkIfAdmin = isAdmin;

createMenu();

prepareGUI();

}

private void createMenu() {

this.mnuItemExit = new JMenuItem("Exit");

this.mnuFile.add(this.mnuItemExit);

this.mnuItemUpdate = new JMenuItem("Update Ticket");

this.mnuAdmin.add(this.mnuItemUpdate);

this.mnuItemDelete = new JMenuItem("Delete Ticket");

this.mnuAdmin.add(this.mnuItemDelete);

this.mnuItemOpenTicket = new JMenuItem("Open Ticket");

this.mnuTickets.add(this.mnuItemOpenTicket);

this.mnuItemViewTicket = new JMenuItem("View Ticket");

this.mnuTickets.add(this.mnuItemViewTicket);

this.mnuItemExit.addActionListener(this);

this.mnuItemUpdate.addActionListener(this);

this.mnuItemDelete.addActionListener(this);

this.mnuItemOpenTicket.addActionListener(this);

this.mnuItemViewTicket.addActionListener(this);

}

private void prepareGUI() {

JMenuBar bar = new JMenuBar();

bar.add(this.mnuFile);

bar.add(this.mnuAdmin);

bar.add(this.mnuTickets);

setJMenuBar(bar);

addWindowListener((WindowListener)new Object(this));

setSize(400, 400);

getContentPane().setBackground(Color.LIGHT\_GRAY);

setLocationRelativeTo(null);

setVisible(true);

}

public void actionPerformed(ActionEvent e) {

if (e.getSource() == this.mnuItemExit) {

System.exit(0);

} else if (e.getSource() == this.mnuItemOpenTicket) {

String ticketName = JOptionPane.showInputDialog((Component)null, "Enter your name");

String ticketDesc = JOptionPane.showInputDialog((Component)null, "Enter a ticket description");

int id = this.dao.insertRecords(ticketName, ticketDesc);

if (id != 0) {

System.out.println("Ticket ID : " + id + " created successfully!!!");

JOptionPane.showMessageDialog(null, "Ticket id: " + id + " created");

} else {

System.out.println("Ticket cannot be created!!!");

}

} else if (e.getSource() == this.mnuItemViewTicket) {

try {

JTable jt = new JTable(ticketsJTable.buildTableModel(this.dao.readRecords()));

jt.setBounds(30, 40, 200, 400);

JScrollPane sp = new JScrollPane(jt);

add(sp);

setVisible(true);

} catch (SQLException e1) {

e1.printStackTrace();

}

}

}

}