

**COMSATS University Islamabad, Lahore Campus**

**Subject**:

Object Oriented Programming.

**Submitted to:**

Sir Shahid Bhatti

**Submitted by**:

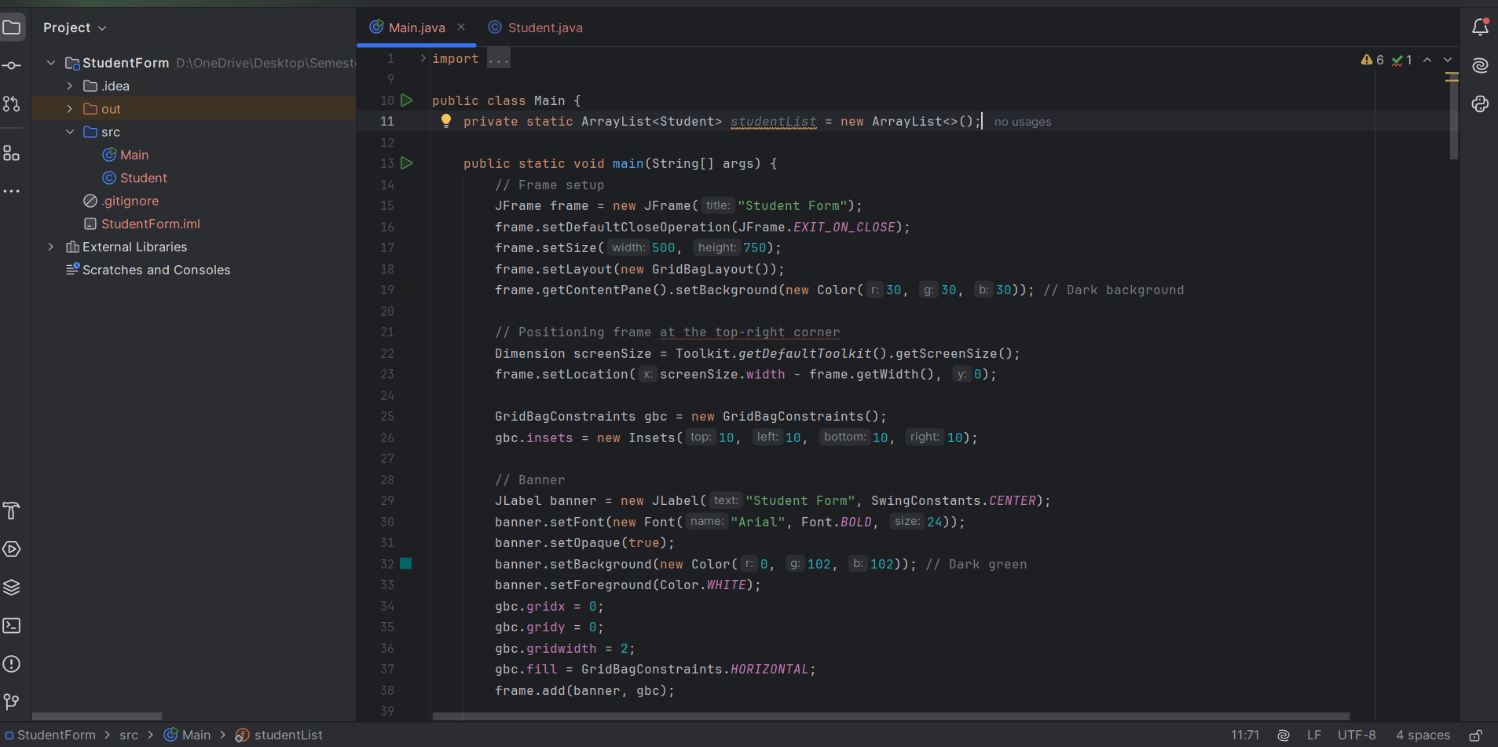
Hamdan Zulfiqar (SP24-BSE-032)

**Section**:

‘A’

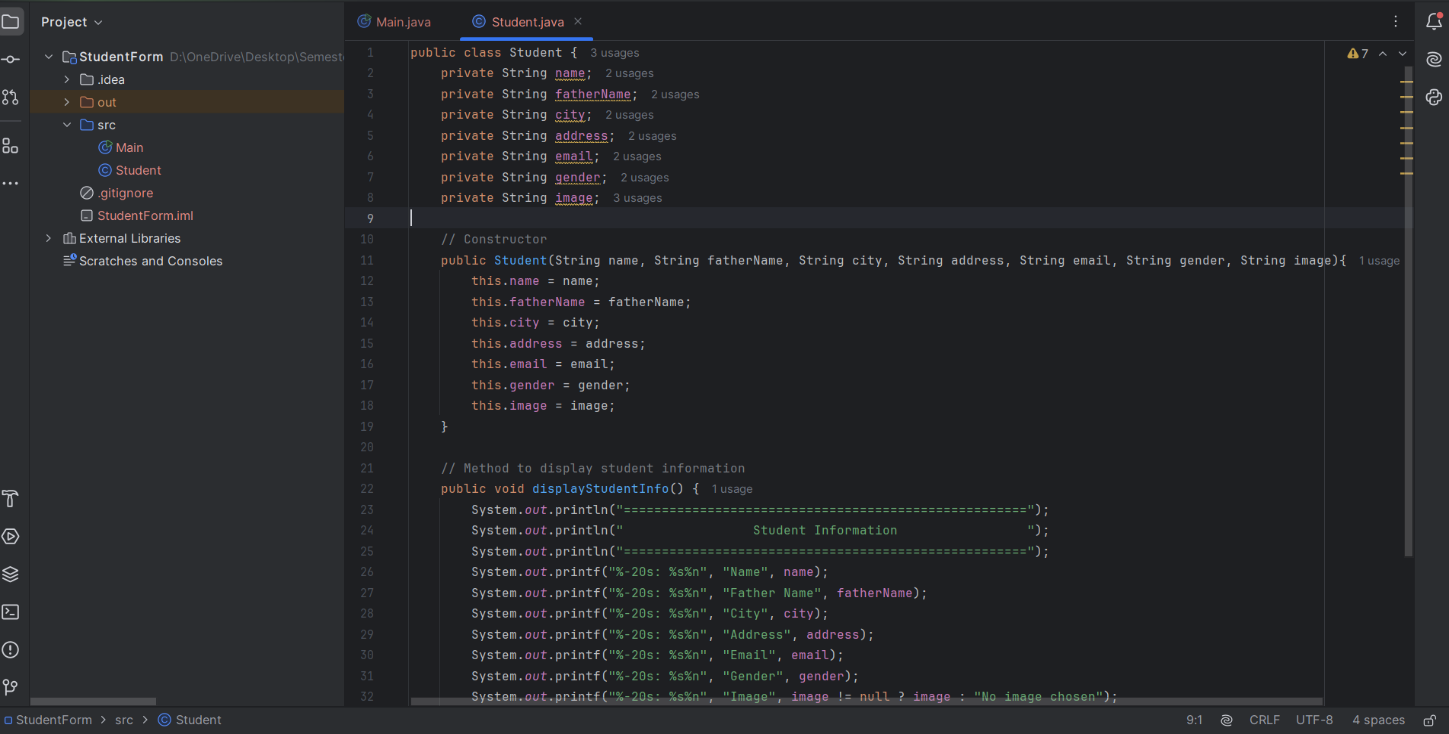
**Student Form.**

**Mian File:**

****

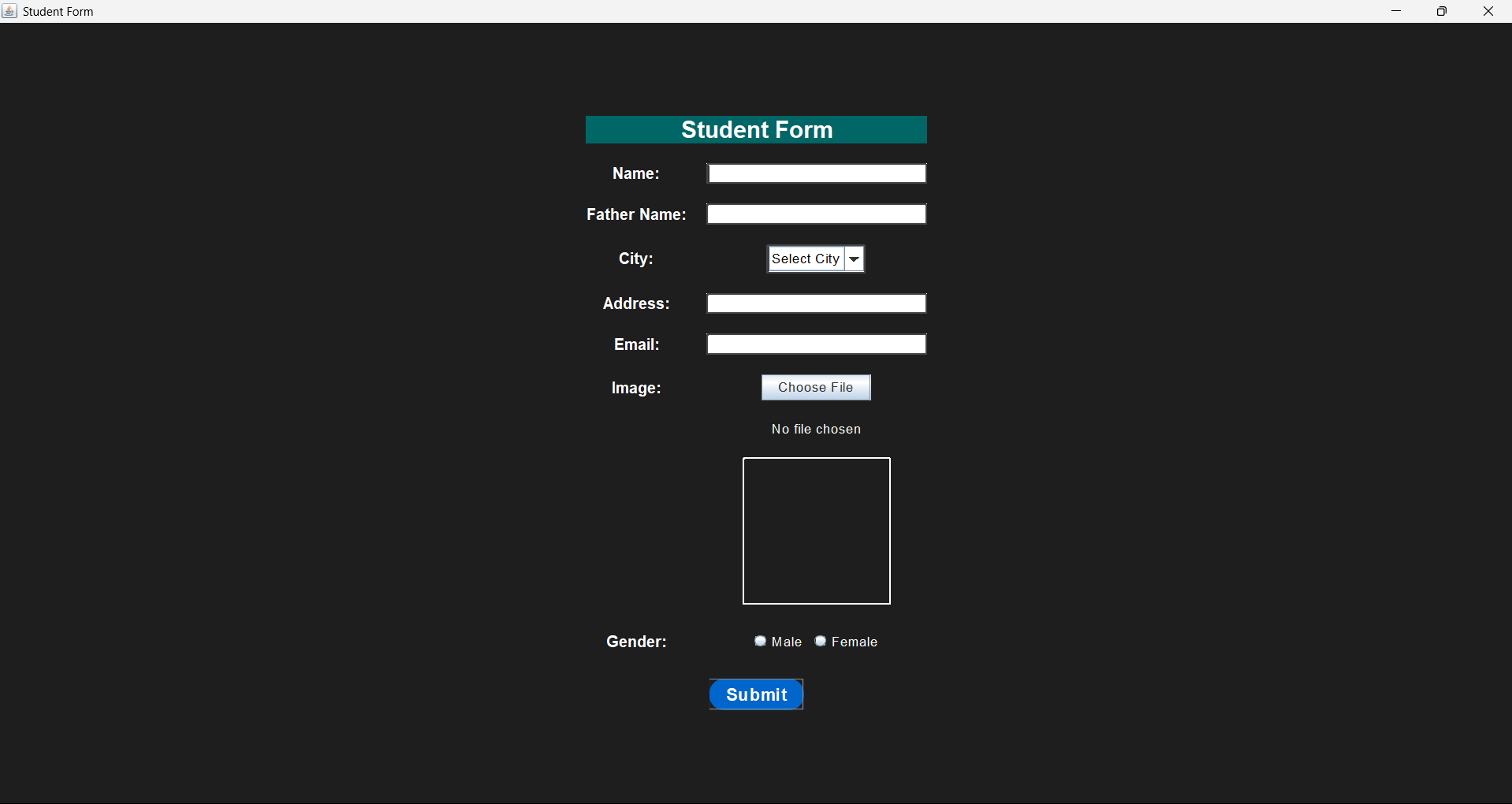
In main.java file, the code creates a graphical user interface **(GUI).** This program makes a simple student registration form with various fields, buttons, and features.

**Student Class:**

****

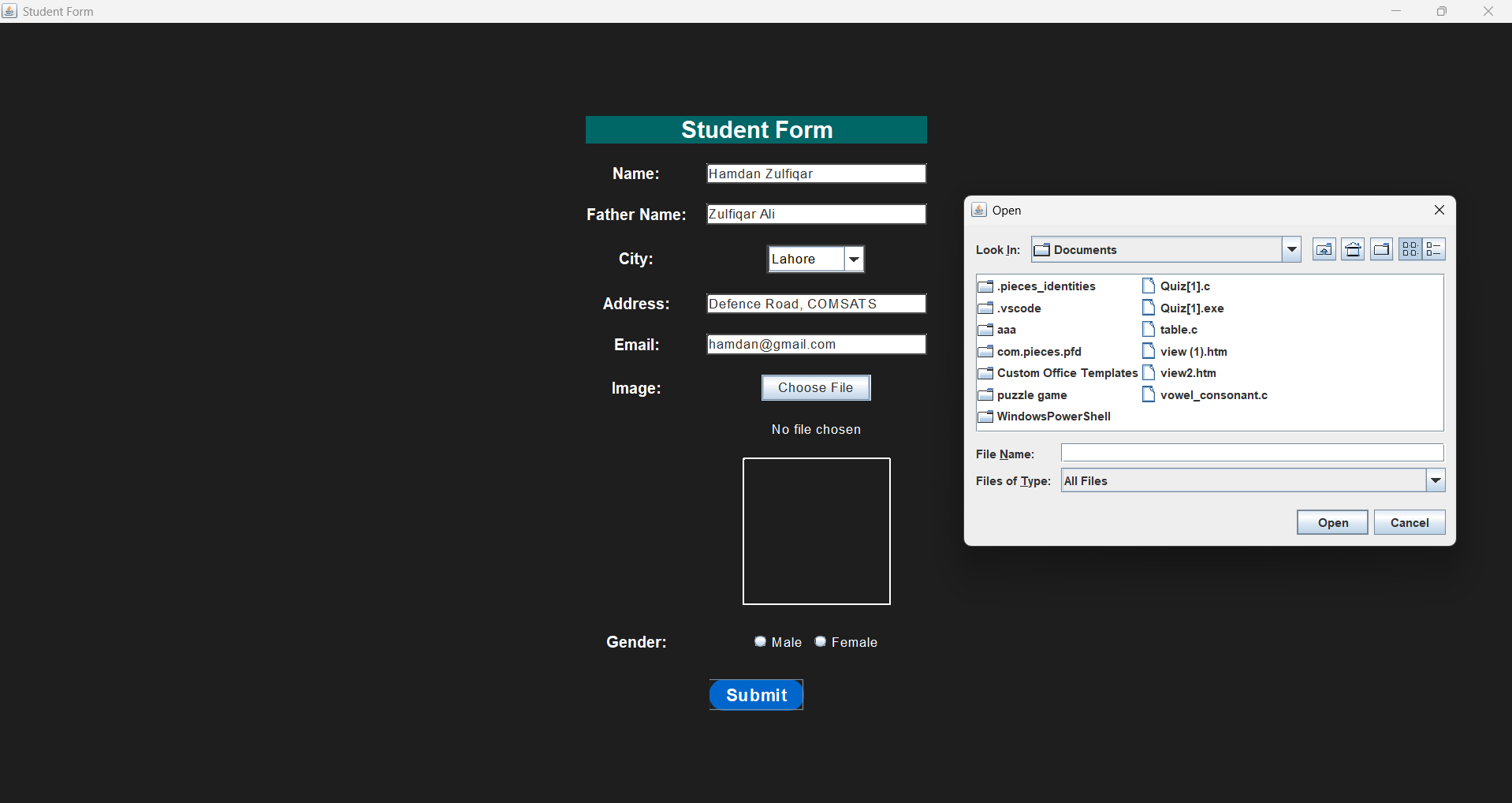
The student class is a simple Java class designed to represent a student's information with key features like attributes, constructor and a display info method to display the student information.

**Display of the Student Form:**

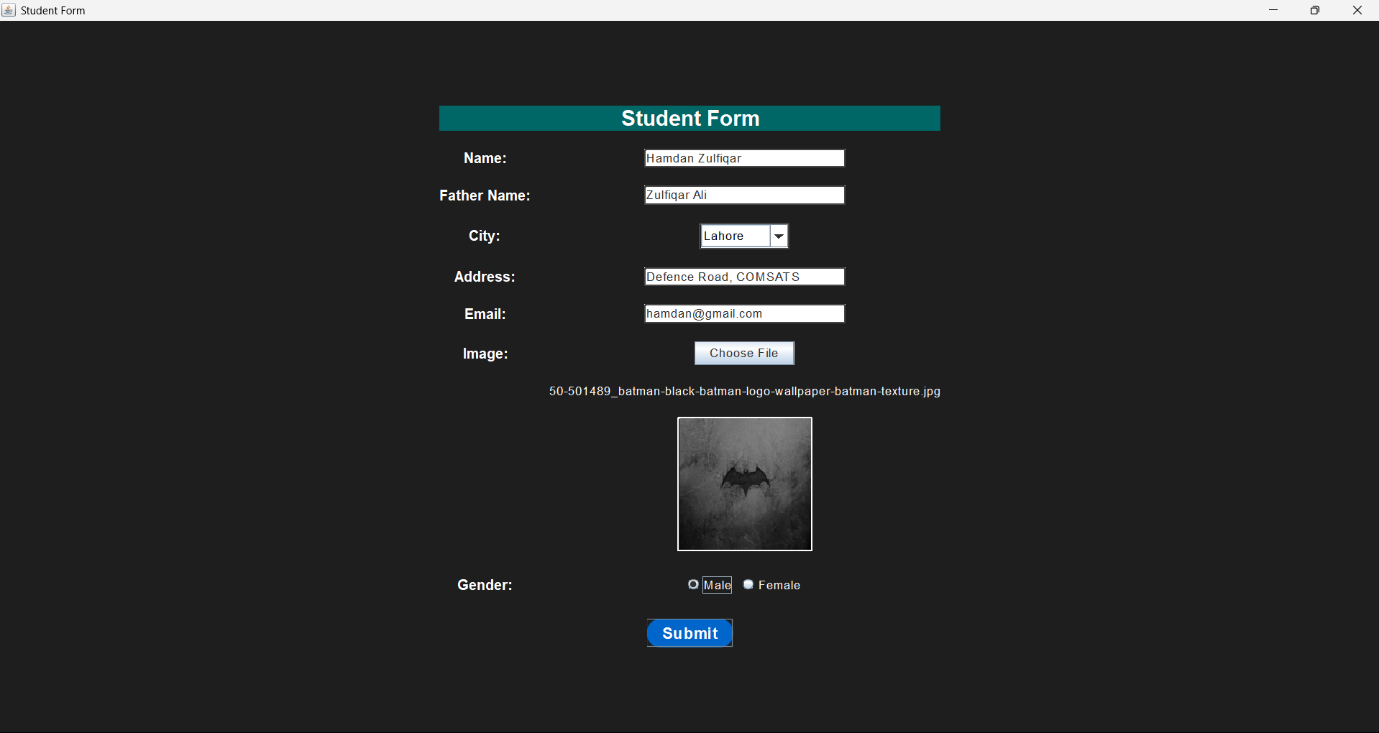
****

In this form, we can enter the name of the student, Father name, the city, address, email of the student and the image chosen by the student, along with gender. After entering the information, Click the submit button to submit the information.

**Filling the Information:**

****

Fill the information, and for the image, click on the **Choose File** option, the file chooser dialog will get opened. Choose the image file you want to add in the form.



After choosing the image file, the image will be displayed on the student form.

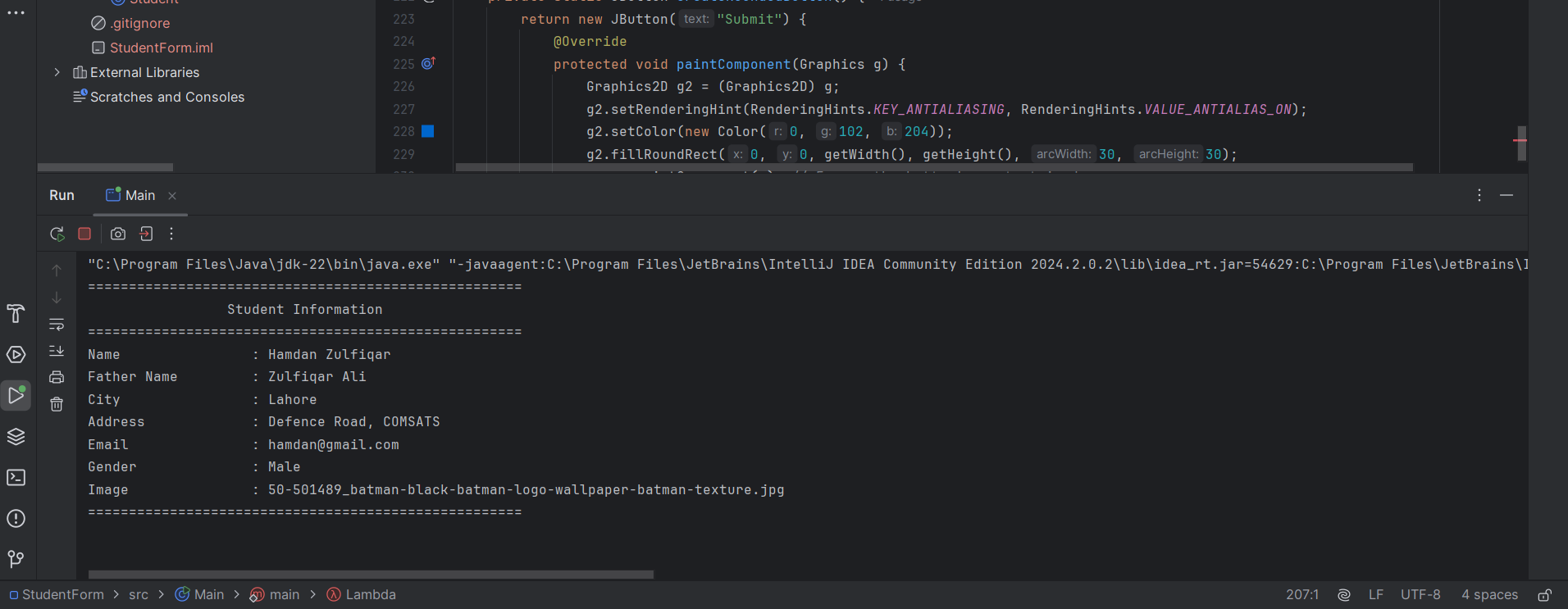
**Submit Button:**

**A screenshot of a computer

Description automatically generated**

Click on the submit button for the data to be submitted successfully in the console.

**Display of Student Info in Console:**

****

After the submission of data, the Student Info will be displayed in the terminal/console.

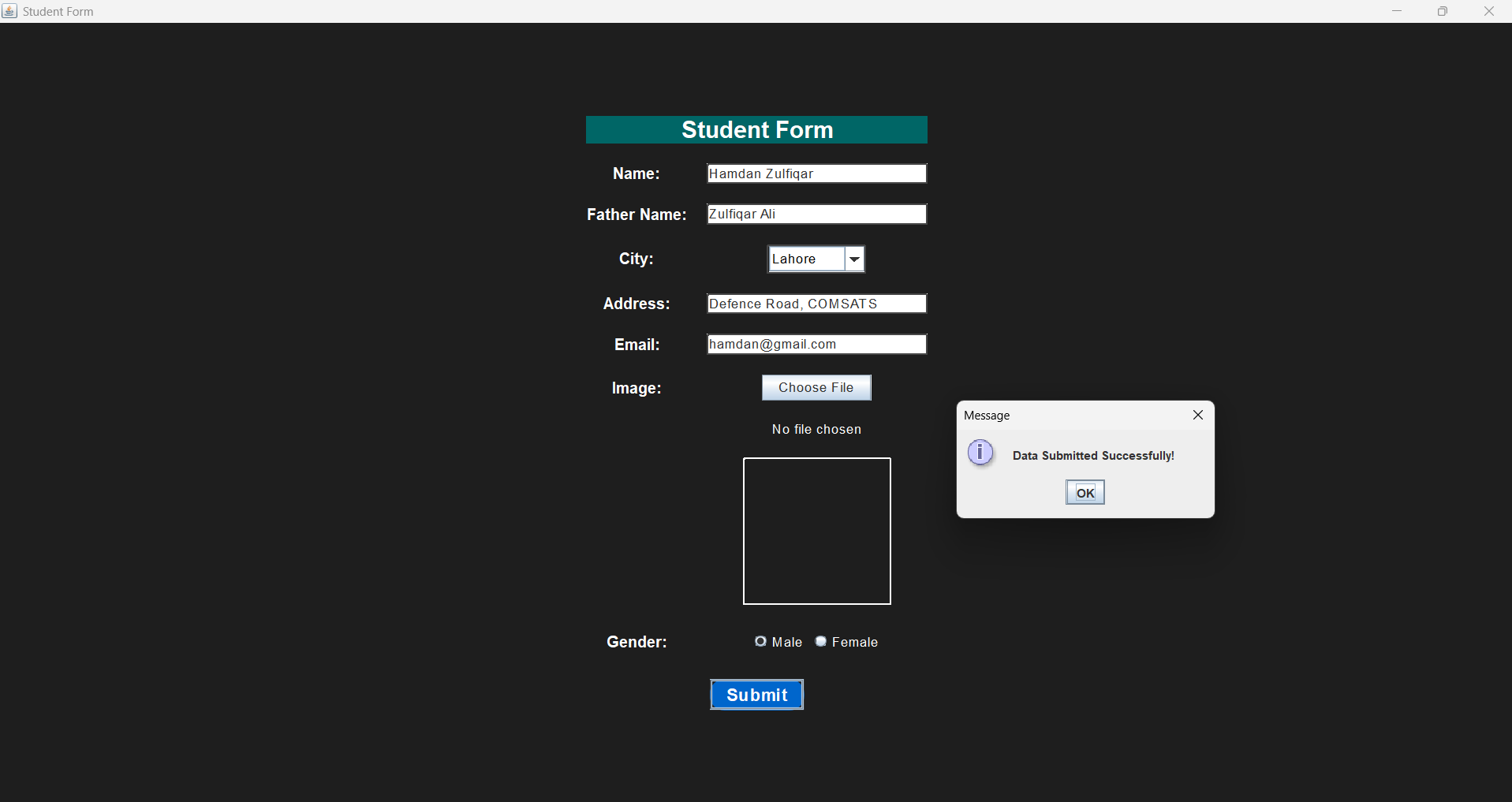
**Submission of data is Compulsory:**

**A computer screen shot of a computer screen

Description automatically generated**

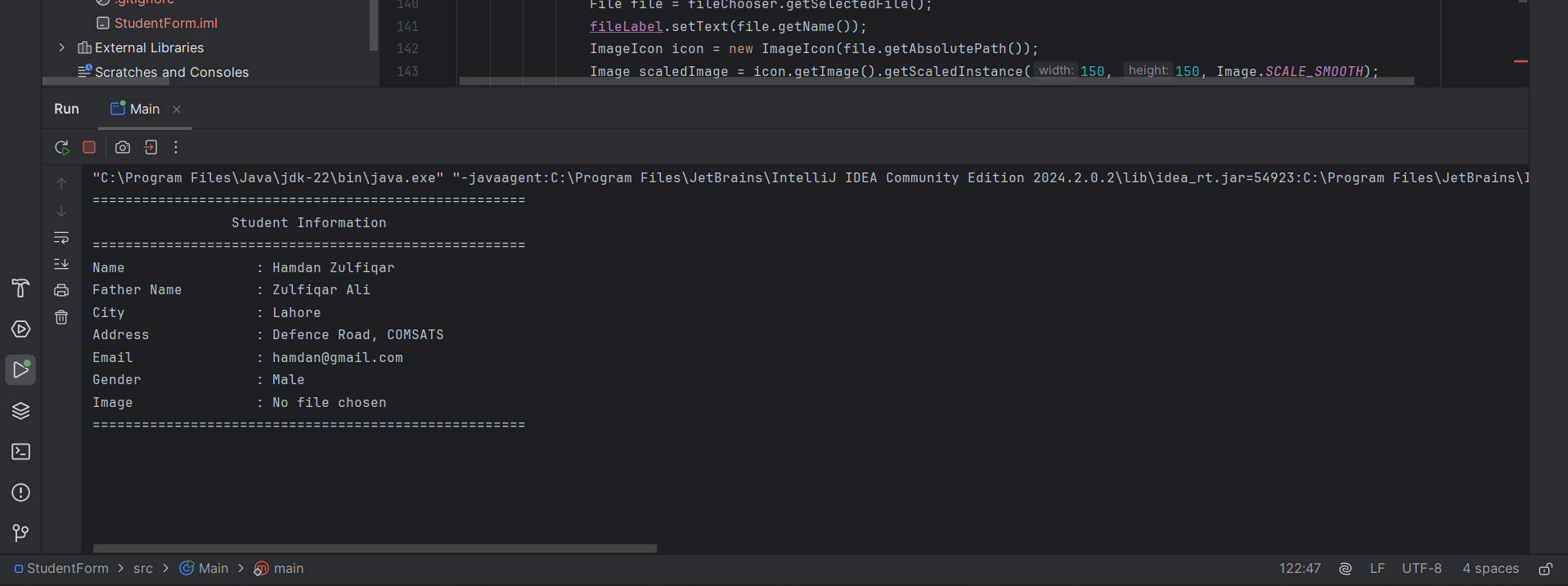
In case of submission of incomplete form, error will be displayed.

**Choosing Image is Optional:**

****

If the image is not selected, the data will successfully get submitted.

The following information will be displayed in the console:



**Source Code:**

import javax.swing.\*;

import javax.swing.border.Border;

import javax.swing.border.LineBorder;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.io.File;

import java.util.ArrayList;

public class Main {

private static ArrayList<Student> studentList = new ArrayList<>();

public static void main(String[] args) {

// Frame setup

JFrame frame = new JFrame("Student Form");

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setSize(500, 750);

frame.setLayout(new GridBagLayout());

frame.getContentPane().setBackground(new Color(30, 30, 30)); // Dark background

// Positioning frame at the top-right corner

Dimension screenSize = Toolkit.getDefaultToolkit().getScreenSize();

frame.setLocation(screenSize.width - frame.getWidth(), 0);

GridBagConstraints gbc = new GridBagConstraints();

gbc.insets = new Insets(10, 10, 10, 10);

// Banner

JLabel banner = new JLabel("Student Form", SwingConstants.CENTER);

banner.setFont(new Font("Arial", Font.BOLD, 24));

banner.setOpaque(true);

banner.setBackground(new Color(0, 102, 102)); // Dark green

banner.setForeground(Color.WHITE);

gbc.gridx = 0;

gbc.gridy = 0;

gbc.gridwidth = 2;

gbc.fill = GridBagConstraints.HORIZONTAL;

frame.add(banner, gbc);

// Labels and TextFields

gbc.gridwidth = 1;

gbc.fill = GridBagConstraints.NONE;

Font labelFont = new Font("Arial", Font.BOLD, 16);

Font fieldFont = new Font("Arial", Font.PLAIN, 14);

Border fieldBorder = new LineBorder(new Color(70, 70, 70), 2, true);

JLabel nameLabel = new JLabel("Name:");

nameLabel.setFont(labelFont);

nameLabel.setForeground(Color.WHITE);

gbc.gridx = 0;

gbc.gridy = 1;

frame.add(nameLabel, gbc);

JTextField nameField = new JTextField(20);

nameField.setFont(fieldFont);

nameField.setBorder(fieldBorder);

gbc.gridx = 1;

frame.add(nameField, gbc);

JLabel fatherLabel = new JLabel("Father Name:");

fatherLabel.setFont(labelFont);

fatherLabel.setForeground(Color.WHITE);

gbc.gridx = 0;

gbc.gridy = 2;

frame.add(fatherLabel, gbc);

JTextField fatherField = new JTextField(20);

fatherField.setFont(fieldFont);

fatherField.setBorder(fieldBorder);

gbc.gridx = 1;

frame.add(fatherField, gbc);

JLabel cityLabel = new JLabel("City:");

cityLabel.setFont(labelFont);

cityLabel.setForeground(Color.WHITE);

gbc.gridx = 0;

gbc.gridy = 3;

frame.add(cityLabel, gbc);

JComboBox<String> cityCombo = new JComboBox<>(new String[]{"Select City", "Lahore", "Islamabad", "Multan", "Karachi"});

cityCombo.setFont(fieldFont);

cityCombo.setBackground(Color.WHITE);

cityCombo.setForeground(Color.BLACK);

JPanel comboPanel = new JPanel(new BorderLayout());

comboPanel.setBorder(fieldBorder);

comboPanel.add(cityCombo);

comboPanel.setBackground(new Color(30, 30, 30));

gbc.gridx = 1;

frame.add(comboPanel, gbc);

JLabel addressLabel = new JLabel("Address:");

addressLabel.setFont(labelFont);

addressLabel.setForeground(Color.WHITE);

gbc.gridx = 0;

gbc.gridy = 4;

frame.add(addressLabel, gbc);

JTextField addressField = new JTextField(20);

addressField.setFont(fieldFont);

addressField.setBorder(fieldBorder);

gbc.gridx = 1;

frame.add(addressField, gbc);

JLabel emailLabel = new JLabel("Email:");

emailLabel.setFont(labelFont);

emailLabel.setForeground(Color.WHITE);

gbc.gridx = 0;

gbc.gridy = 5;

frame.add(emailLabel, gbc);

JTextField emailField = new JTextField(20);

emailField.setFont(fieldFont);

emailField.setBorder(fieldBorder);

gbc.gridx = 1;

frame.add(emailField, gbc);

JLabel imageLabel = new JLabel("Image:");

imageLabel.setFont(labelFont);

imageLabel.setForeground(Color.WHITE);

gbc.gridx = 0;

gbc.gridy = 6;

frame.add(imageLabel, gbc);

JButton fileChooserButton = new JButton("Choose File");

fileChooserButton.setFont(fieldFont);

JLabel fileLabel = new JLabel("No file chosen");

fileLabel.setFont(fieldFont);

fileLabel.setForeground(Color.WHITE);

JLabel imagePreview = new JLabel();

imagePreview.setPreferredSize(new Dimension(150, 150));

imagePreview.setBorder(new LineBorder(Color.WHITE, 2, true));

fileChooserButton.addActionListener(e -> {

JFileChooser fileChooser = new JFileChooser();

if (fileChooser.showOpenDialog(frame) == JFileChooser.APPROVE\_OPTION) {

File file = fileChooser.getSelectedFile();

fileLabel.setText(file.getName());

ImageIcon icon = new ImageIcon(file.getAbsolutePath());

Image scaledImage = icon.getImage().getScaledInstance(150, 150, Image.SCALE\_SMOOTH);

imagePreview.setIcon(new ImageIcon(scaledImage));

}

});

gbc.gridx = 1;

frame.add(fileChooserButton, gbc);

gbc.gridy = 7;

frame.add(fileLabel, gbc);

gbc.gridy = 8;

gbc.gridwidth = 2;

frame.add(imagePreview, gbc);

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

genderLabel.setFont(labelFont);

genderLabel.setForeground(Color.WHITE);

gbc.gridx = 0;

gbc.gridy = 9;

gbc.gridwidth = 1;

frame.add(genderLabel, gbc);

JRadioButton maleButton = new JRadioButton("Male");

maleButton.setFont(fieldFont);

maleButton.setForeground(Color.WHITE);

maleButton.setBackground(new Color(30, 30, 30));

JRadioButton femaleButton = new JRadioButton("Female");

femaleButton.setFont(fieldFont);

femaleButton.setForeground(Color.WHITE);

femaleButton.setBackground(new Color(30, 30, 30));

ButtonGroup genderGroup = new ButtonGroup();

genderGroup.add(maleButton);

genderGroup.add(femaleButton);

JPanel genderPanel = new JPanel();

genderPanel.add(maleButton);

genderPanel.add(femaleButton);

genderPanel.setBackground(new Color(30, 30, 30));

gbc.gridx = 1;

frame.add(genderPanel, gbc);

JButton submitButton = createRoundedButton();

submitButton.setFont(new Font("Arial", Font.BOLD, 18));

submitButton.setForeground(Color.WHITE);

submitButton.setFocusPainted(false);

submitButton.setContentAreaFilled(false);

gbc.gridx = 0;

gbc.gridy = 10;

gbc.gridwidth = 2;

frame.add(submitButton, gbc);

submitButton.addActionListener(e -> {

String name = nameField.getText();

String fatherName = fatherField.getText();

String city = cityCombo.getSelectedItem() != null ? (String) cityCombo.getSelectedItem() : "Not selected";

String address = addressField.getText();

String email = emailField.getText();

String gender = maleButton.isSelected() ? "Male" : femaleButton.isSelected() ? "Female" : "Not selected";

String image = fileLabel.getText();

if (name.isEmpty() || city.equals("Select City")) {

JOptionPane.showMessageDialog(frame, "Please fill all required fields!", "Error", JOptionPane.ERROR\_MESSAGE);

return;

}

Student student = new Student(name, fatherName, city, address, email, gender, image);

studentList.add(student);

JOptionPane.showMessageDialog(frame, "Data Submitted Successfully!");

student.displayStudentInfo();

});

frame.setVisible(true);

}

private static JButton createRoundedButton() {

return new JButton("Submit") {

@Override

protected void paintComponent(Graphics g) {

Graphics2D g2 = (Graphics2D) g;

g2.setRenderingHint(RenderingHints.KEY\_ANTIALIASING, RenderingHints.VALUE\_ANTIALIAS\_ON);

g2.setColor(new Color(0, 102, 204));

g2.fillRoundRect(0, 0, getWidth(), getHeight(), 30, 30);

super.paintComponent(g);

}

};

}

static class Student {

private String name;

private String fatherName;

private String city;

private String address;

private String email;

private String gender;

private String image;

public Student(String name, String fatherName, String city, String address, String email, String gender, String image) {

this.name = name;

this.fatherName = fatherName;

this.city = city;

this.address = address;

this.email = email;

this.gender = gender;

this.image = image;

}

public void displayStudentInfo() {

System.out.println("=====================================================");

System.out.println(" Student Information ");

System.out.println("=====================================================");

System.out.printf("%-20s: %s%n", "Name", name);

System.out.printf("%-20s: %s%n", "Father Name", fatherName);

System.out.printf("%-20s: %s%n", "City", city);

System.out.printf("%-20s: %s%n", "Address", address);

System.out.printf("%-20s: %s%n", "Email", email);

System.out.printf("%-20s: %s%n", "Gender", gender);

System.out.printf("%-20s: %s%n", "Image", image != null ? image : "No image chosen");

System.out.println("=====================================================");

}

}

}

**THE END.**