

Student Name:	Mandar Chaudhari
Reg No	220968222
Assignment No.	FISAC
Subject Code	DSE 2123
Subject	OOP with Java
Marks	10M

Java Swing Application

“Admission Test”

Overview

This Java Swing application, named "Admission Test," is designed to facilitate an admission test environment with a graphical user interface created using Java Swings. The application includes features such as a custom-styled login window, multiple-choice question (MCQ) page, and a score page. It incorporates various swing components for GUI design to make the application work perfectly without errors as well as give the user a beautiful GUI experience.

Problem Statement Overview

Create a GUI application in Java for conducting an admission entrance test consisting of Multiple-Choice Questions (MCQs) with the following requirements:

1. Login Window:

- The application should open a login window for user authentication.
- Users need to enter their credentials (username and password) for validation.
- After successful login, the application should navigate to a new page.
- This new page should contain a set of multiple-choice questions (MCQs).

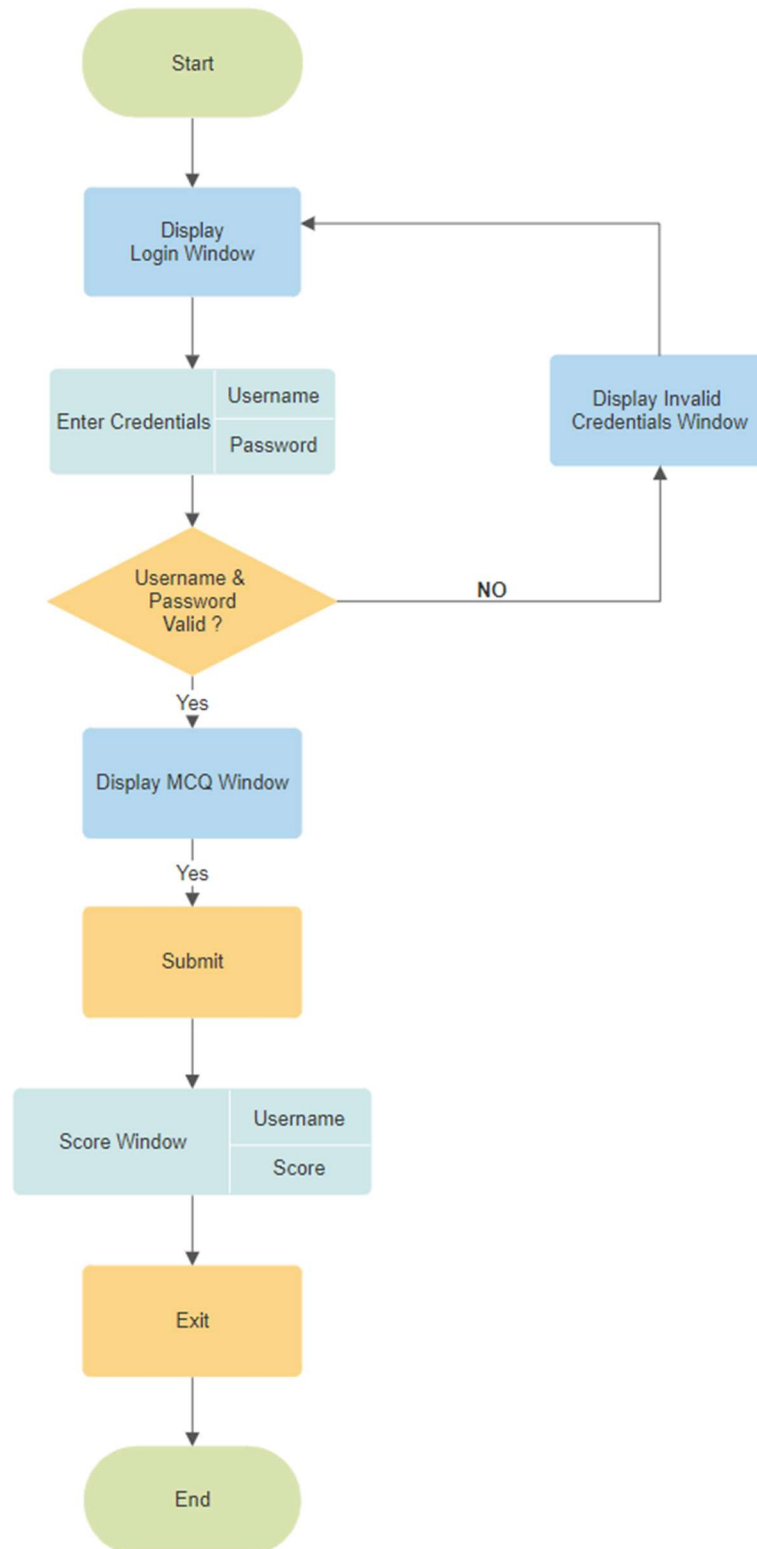
2. MCQ Page:

- Display 5 MCQs on this page, each with a question and multiple radio buttons for choosing options.
- Users should be able to select one option per question.
- Provide a "Submit" button for users to submit their answers.

3. Result Display:

- After the user has attempted all the questions and submitted their answers, the application should calculate and display the total score.

Flow Diagram



Components Used

1. **JFrame:** It is used to create the main application window and secondary frames (login, MCQ, score pages).
2. **JPanel:** It is employed for organizing and structuring components within frames. It's used for both the login window, MCQ page, and score page.
3. **JLabel:** It is used to display text, including labels for username, password, error messages, and score information.
4. **TextField:** It is used for user input, specifically for entering the username.
5. **JPasswordField:** It is used for secure password input during the login process.
6. **Button:** It is utilized for creating interactive buttons such as the login button, OK button in error dialogs, submit button on the MCQ page, and exit button on the score page.
7. **JDialog:** It is used to create custom dialogs for displaying error messages and score information.
8. **TextArea:** It is used for displaying multiple-choice questions and error messages with a larger text area.
9. **JRadioButton:** Its components are employed for presenting multiple-choice options to the user.
10. **ButtonGroup:** It is used to group radio buttons for each question, ensuring exclusive selection.
11. **Font & Color:** Custom fonts and colors are set for various components to enhance the visual of the application.

Events & Actions

1. Login Button Click Event:

- Event: `ActionEvent`
- Action: Validates user credentials. If valid, hides the login frame and creates the MCQ page; otherwise, shows a wrong password dialog.

2. OK Button Click Event in Wrong Password Dialog:

- Event: `ActionEvent`
- Action: Closes the wrong password dialog when the "OK" button is clicked.

3. Submit Button Click Event on MCQ Page:

- Event: `ActionEvent`
- Action: Calculates and checks the number of correct answers when the user clicks the "Submit" button. Then, displays the score page.

4. Exit Button Click Event on Score Page:

- Event: `ActionEvent`
- Action: Exits the application when the "Exit" button is clicked on the score page.

Program code with comments:

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.util.Enumeraation;

public class AppMine {
    // Dummy credentials
    private static final String USERNAME_MAIN = "220968222";
    private static final String PASSWORD_MAIN = "mandar";

    private JFrame loginFrame;
    private JFrame mcqFrame;
    private JTextField usernameField;
    private JPasswordField passwordField;
    private String currentUser;

    public static void main(String[] args) {
        SwingUtilities.invokeLater(() -> new AppMine().createLoginWindow());
    }

    // create a login window
    private void createLoginWindow() {
        loginFrame = new JFrame("Admission Test Login");
        loginFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        JPanel panel = new JPanel(new GridLayout(4, 2, 0, 20)) {
            @Override
            protected void paintComponent(Graphics g) {
                super.paintComponent(g);

                Color color1 = new Color(169, 241, 223);
                Color color2 = new Color(255, 187, 187);

                GradientPaint gradient = new GradientPaint(0, 0, color1,
getWidth(), getHeight(), color2);

                Graphics2D g2d = (Graphics2D) g;

                g2d.setPaint(gradient);

                g2d.fillRect(0, 0, getWidth(), getHeight());
            }
        };
        panel.setBorder(BorderFactory.createEmptyBorder(60, 60, 60, 60)); //
Doubled the padding

        JLabel usernameLabel = new JLabel("U s e r n a m e:");
        JLabel passwordLabel = new JLabel("P a s s w o r d:");
        usernameField = new JTextField();
        passwordField = new JPasswordField();
    }
}
```

```

        JButton loginButton = new JButton("L O G I N");

        Font textFieldFont = new Font("Agency FB", Font.BOLD, 30);
        usernameField.setBorder(BorderFactory.createLineBorder(Color.GRAY,
2));
        usernameField.setPreferredSize(new Dimension(300, 50));
        usernameField.setFont(textFieldFont);
        usernameField.setForeground(Color.DARK_GRAY);
        usernameField.setHorizontalAlignment(JTextField.CENTER);

        passwordField.setBorder(BorderFactory.createLineBorder(Color.GRAY,
2));
        passwordField.setPreferredSize(new Dimension(300, 50));
        passwordField.setFont(textFieldFont);
        passwordField.setForeground(Color.DARK_GRAY);
        passwordField.setHorizontalAlignment(JTextField.CENTER);

        Font labelFont = new Font("Agency FB", Font.BOLD, 40);
        usernameLabel.setFont(labelFont);

        passwordLabel.setFont(labelFont);

        loginButton.setBorder(BorderFactory.createLineBorder(Color.DARK_GRAY,
4));
        loginButton.setBackground(new Color(100, 174, 152));
        loginButton.setForeground(Color.WHITE);
        loginButton.setFont(new Font("Agency FB", Font.BOLD, 28));

        panel.add(usernameLabel);
        panel.add(usernameField);
        panel.add(passwordLabel);
        panel.add(passwordField);
        panel.add(new JLabel());
        panel.add(loginButton);

        loginButton.addActionListener(new ActionListener() {
            @Override
            public void actionPerformed(ActionEvent e) {
                if (validateCredentials()) {
                    loginFrame.setVisible(false);
                    createMCQPage(usernameField.getText());
                } else {
                    showWrongPasswordDialog();
                }
            }
        });

        loginFrame.getContentPane().add(BorderLayout.CENTER, panel);
        loginFrame.setSize(800, 400); // Doubled the frame size
        loginFrame.setLocationRelativeTo(null);
        loginFrame.setVisible(true);
    }

    // method to show a custom wrong password dialog
    private void showWrongPasswordDialog() {
        JDialog wrongPasswordDialog = new JDialog(loginFrame, "Error", true);
        JPanel dialogPanel = new JPanel(new GridLayout(3, 1, 0, 20)) {

```

```

@Override
protected void paintComponent(Graphics g) {
    super.paintComponent(g);

    Color color1 = new Color(169, 241, 223);
    Color color2 = new Color(255, 187, 187);
    GradientPaint gradient = new GradientPaint(0, 0, color1,
getWidth(), getHeight(), color2);
    Graphics2D g2d = (Graphics2D) g;
    g2d.setPaint(gradient);
    g2d.fillRect(0, 0, getWidth(), getHeight());
}
};
dialogPanel.setBorder(BorderFactory.createEmptyBorder(60, 60, 60,
60));

JLabel errorMessageLabel = new JLabel("Invalid Credentials. Please
Try Again.");
errorMessageLabel.setFont(new Font("Agency FB", Font.BOLD, 30));

JButton okButton = new JButton("OK");

okButton.setBorder(BorderFactory.createLineBorder(Color.DARK_GRAY,
4));
okButton.setBackground(new Color(100, 174, 152));
okButton.setForeground(Color.WHITE);
okButton.setFont(new Font("Agency FB", Font.BOLD, 28));

okButton.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent e) {
    wrongPasswordDialog.dispose();
}
});

dialogPanel.add(errorMessageLabel);
dialogPanel.add(new JLabel());
dialogPanel.add(okButton);

wrongPasswordDialog.getContentPane().add(BorderLayout.CENTER,
dialogPanel);
wrongPasswordDialog.setSize(500, 300);
wrongPasswordDialog.setLocationRelativeTo(loginFrame);
wrongPasswordDialog.setVisible(true);
}

// Code to verify credentials
private boolean validateCredentials() {
    String enteredUsername = usernameField.getText();
    char[] enteredPassword = passwordField.getPassword();
    String enteredPasswordString = new String(enteredPassword);
    return enteredUsername.equals(USERNAME_MAIN) &&
enteredPasswordString.equals(PASSWORD_MAIN);
}

// Code to create a MCQ page
private void createMCQPage(String username) {

```



```

currentUser = username; // Store the username

mcqFrame = new JFrame("MCQ Page for " + username);
mcqFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

JPanel mcqPanel = new JPanel(new GridLayout(5, 1, 10, 15)) {
    @Override
    protected void paintComponent(Graphics g) {
        super.paintComponent(g);

        Color color1 = new Color(169, 241, 223);
        Color color2 = new Color(255, 187, 187);
        GradientPaint gradient = new GradientPaint(0, 0, color1,
getWidth(), getHeight(), color2);
        Graphics2D g2d = (Graphics2D) g;
        g2d.setPaint(gradient);
        g2d.fillRect(0, 0, getWidth(), getHeight());
    }
};
mcqPanel.setBorder(BorderFactory.createEmptyBorder(20, 20, 20, 20));

createQuestionPanel(mcqPanel, "1. Which of these are types of
multitasking?", "A. Process based", "B. Thread based", "C. Process and Thread
based", 3);
createQuestionPanel(mcqPanel, "2. Thread priority in Java is?", "A.
Integer", "B. Float", "C. Double", 1);
createQuestionPanel(mcqPanel, "3. Which class is used to display text
in Java Swing?", "A. JTextField", "B. JLabel", "C. JList", 2);
createQuestionPanel(mcqPanel, "4. Which method is used to set the
text content of a JTextField?", "A. setText()", "B. setContent()", "C.
setLabel()", 1);
createQuestionPanel(mcqPanel, "5. How many interfaces can a single
class implement?", "A. One", "B. Two", "C. Unlimited", 3);

JButton submitButton = new JButton("Submit");

submitButton.setBorder(BorderFactory.createLineBorder(Color.DARK_GRAY, 3));
submitButton.setBackground(new Color(100, 174, 152));
submitButton.setForeground(Color.WHITE);
submitButton.setFont(new Font("Agency FB", Font.BOLD, 28));

submitButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        int correctAnswers = 0;

        if (isSelectedCorrect(optionGroups[0], 2)) {
            correctAnswers++;
        }
        if (isSelectedCorrect(optionGroups[1], 0)) {
            correctAnswers++;
        }
        if (isSelectedCorrect(optionGroups[2], 1)) {
            correctAnswers++;
        }
        if (isSelectedCorrect(optionGroups[3], 0)) {

```

```

        correctAnswers++;
    }
    if (isSelectedCorrect(optionGroups[4], 2)) {
        correctAnswers++;
    }

    showScorePage(username, correctAnswers);

    }
});

mcqFrame.getContentPane().add(BorderLayout.CENTER, mcqPanel);
mcqFrame.add(submitButton, BorderLayout.SOUTH);

JLabel usernameLabel = new JLabel(" WELCOME , " + username + " !");
usernameLabel.setFont(new Font("Agency FB", Font.BOLD, 20));
mcqFrame.add(usernameLabel, BorderLayout.NORTH);

mcqFrame.setSize(800, 720);
mcqFrame.setLocationRelativeTo(null); // Center the frame
mcqFrame.setVisible(true);
}

// method to display the score in a new window
private void showScorePage(String username, int correctAnswers) {
    JFrame scoreFrame = new JFrame("Score Page");
    scoreFrame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    JPanel panel = new JPanel(new GridLayout(4, 1, 0, 20)) {
        @Override
        protected void paintComponent(Graphics g) {
            super.paintComponent(g);

            Color color1 = new Color(169, 241, 223);
            Color color2 = new Color(255, 187, 187);
            GradientPaint gradient = new GradientPaint(0, 0, color1,
getWidth(), getHeight(), color2);
            Graphics2D g2d = (Graphics2D) g;
            g2d.setPaint(gradient);
            g2d.fillRect(0, 0, getWidth(), getHeight());
        }
    };
    panel.setBorder(BorderFactory.createEmptyBorder(60, 60, 60, 60));

    JLabel user = new JLabel("Hi " + username + " !");
    user.setFont(new Font("Agency FB", Font.BOLD, 40));
    JLabel scoreLabel = new JLabel("Your Score: " + correctAnswers + " /
5");
    scoreLabel.setFont(new Font("Agency FB", Font.BOLD, 40));

    JButton exitButton = new JButton("Exit");

    exitButton.setBorder(BorderFactory.createLineBorder(Color.DARK_GRAY,
4));
    exitButton.setBackground(new Color(100, 174, 152));
    exitButton.setForeground(Color.WHITE);
    exitButton.setFont(new Font("Agency FB", Font.BOLD, 28));

```

```

exitButton.addActionListener(new ActionListener() {
    @Override
    public void actionPerformed(ActionEvent e) {
        System.exit(0);
    }
});

panel.add(user);
panel.add(scoreLabel);
panel.add(new JLabel());
panel.add(exitButton);

scoreFrame.getContentPane().add(BorderLayout.CENTER, panel);
scoreFrame.setSize(800, 400);
scoreFrame.setLocationRelativeTo(null);
scoreFrame.setVisible(true);
}

// method to create a question panel
private void createQuestionPanel(JPanel mcqPanel, String question, String
option1, String option2, String option3, int correctOption) {
    JPanel questionPanel = new JPanel(new GridLayout(6, 1));

    questionPanel.setBorder(BorderFactory.createLineBorder(Color.GRAY,
3));

    JTextArea questionTextArea = new JTextArea(question);
    questionTextArea.setEditable(false);

    Font textAreaFont = new Font("Consolas", Font.PLAIN, 18);
    questionTextArea.setFont(textAreaFont);

    questionTextArea.setBackground(new Color(240, 240, 240));

    questionTextArea.setBorder(BorderFactory.createLineBorder(Color.GRAY,
0));

    JLabel space = new JLabel();
    JRadioButton option1Button = new JRadioButton(option1);
    JRadioButton option2Button = new JRadioButton(option2);
    JRadioButton option3Button = new JRadioButton(option3);

    Font radioButtonFont = new Font("Consolas", Font.PLAIN, 16);
    option1Button.setFont(radioButtonFont);
    option2Button.setFont(radioButtonFont);
    option3Button.setFont(radioButtonFont);

    ButtonGroup group = new ButtonGroup();
    group.add(option1Button);
    group.add(option2Button);
    group.add(option3Button);

    optionGroups[questionIndex] = group;

    questionPanel.add(questionTextArea);
    questionPanel.add(space);

```

```

        questionPanel.add(option1Button);
        questionPanel.add(option2Button);
        questionPanel.add(option3Button);

        mcqPanel.add(questionPanel);
        questionIndex++;
    }

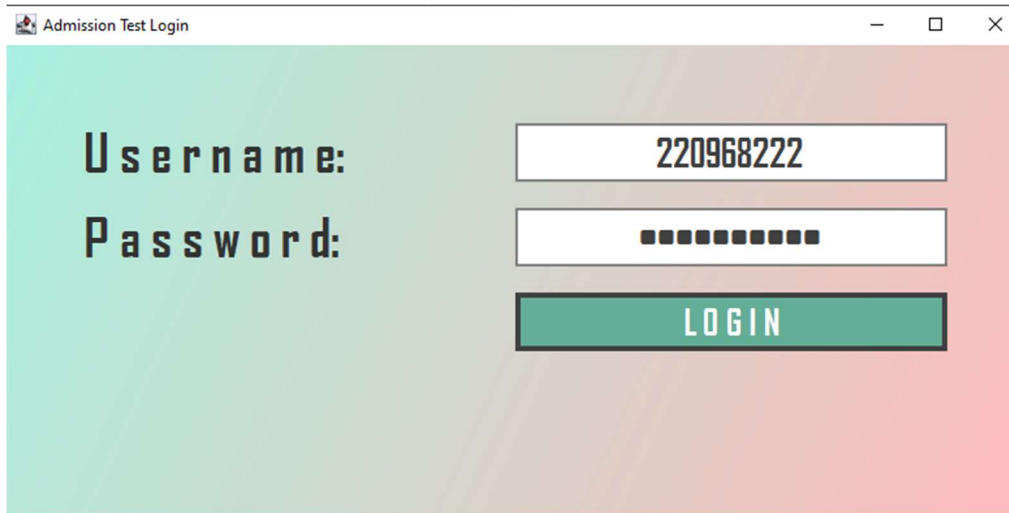
    // method to check if the selected option is correct
    private boolean isSelectedCorrect(ButtonGroup group, int correctIndex) {
        Enumeration<AbstractButton> buttons = group.getElements();
        int selectedIndex = -1;
        for (int i = 0; buttons.hasMoreElements(); i++) {
            AbstractButton button = buttons.nextElement();
            if (button.isSelected()) {
                selectedIndex = i;
                break;
            }
        }
        return selectedIndex == correctIndex;
    }

    private ButtonGroup[] optionGroups = new ButtonGroup[5];
    private int questionIndex = 0;
}

```

Screenshot Of Output

1. Login Page



A screenshot of a web application window titled "Admission Test Login". The window has a light blue header bar with the title and standard window controls (minimize, maximize, close). The main content area has a light blue background. On the left, the labels "Username:" and "Password:" are displayed in a large, bold, black font. To the right of these labels are two white input fields with black borders. The first input field contains the text "220968222". The second input field contains ten black dots, indicating a masked password. Below the input fields is a green button with the text "LOGIN" in white, uppercase letters.

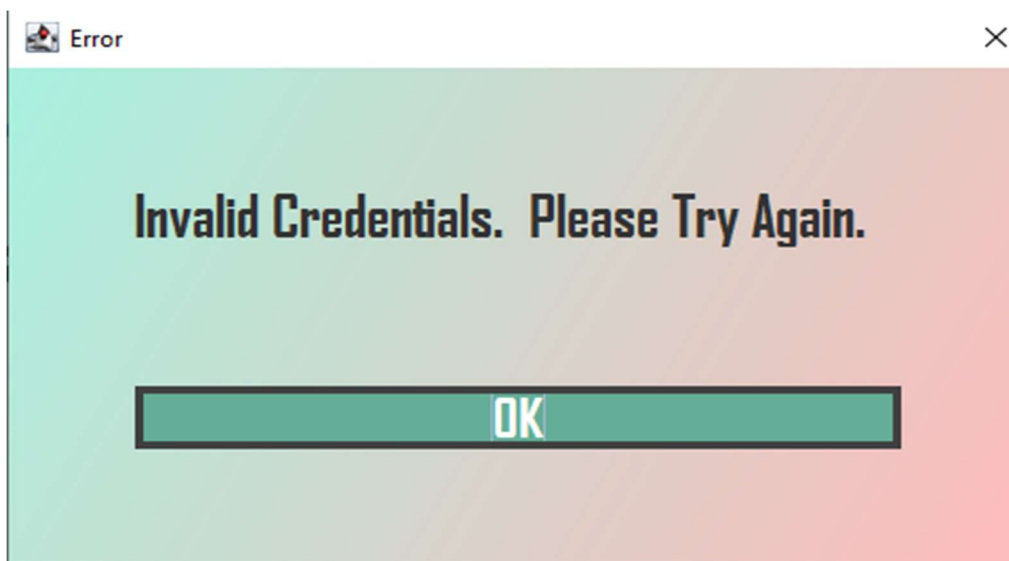
Admission Test Login

Username: 220968222

Password:

LOGIN

2. Invalid Password Page



A screenshot of an error dialog box titled "Error". The dialog has a light blue header bar with the title and a close button (X). The main content area has a light blue background. In the center, the text "Invalid Credentials. Please Try Again." is displayed in a large, bold, black font. Below this text is a green button with the text "OK" in white, uppercase letters.

Error

Invalid Credentials. Please Try Again.

OK

3. MCQ Questions Page

MCQ Page for 220968222

WELCOME, 220968222!

1. Which of these are types of multitasking?

☐ A. Process based
☐ B. Thread based
☒ C. Process and Thread based

2. Thread priority in Java is?

☒ A. Integer
☐ B. Float
☐ C. Double

3. Which class is used to display text in Java Swing?

☐ A. JTextField
☒ B. JLabel
☐ C. JList

4. Which method is used to set the text content of a JTextField?

☒ A. setText()
☐ B. setContent()
☐ C. setLabel()

5. How many interfaces can a single class implement?

☐ A. One
☐ B. Two
☒ C. Unlimited

Submit

4. Score Page

Score Page

Hi 220968222!

Your Score: 5 / 5

Exit

References

<https://www.javatpoint.com/java-swing>

<https://docs.oracle.com/javase/tutorial/uiswing/>

<https://www.geeksforgeeks.org/introduction-to-java-swing/>

<https://www.youtube.com/watch?v=Kmgo00avvEw>