**ASSIGNMENT 9**

**SET A**

**1. Write a java program to design a following GUI. Use appropriate Layout and Components.**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class GUILayout {

public static void main(String[] args) {

// Create the frame for the login window

JFrame frame = new JFrame("Login Page");

// Set the layout for the frame (GridLayout for organized layout)

frame.setLayout(new GridLayout(4, 2, 10, 10)); // 4 rows, 2 columns with padding

// Set the default close operation

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

// Set the size of the frame

frame.setSize(400, 250);

// Create components

JLabel usernameLabel = new JLabel("Username:");

JTextField usernameField = new JTextField(15);

JLabel passwordLabel = new JLabel("Password:");

JPasswordField passwordField = new JPasswordField(15);

JButton loginButton = new JButton("Login");

JButton resetButton = new JButton("Reset");

JLabel messageLabel = new JLabel(""); // Will display messages

// Create a JPanel to hold the buttons (Login and Reset)

JPanel buttonPanel = new JPanel();

buttonPanel.setLayout(new FlowLayout()); // FlowLayout to arrange buttons side by side

buttonPanel.add(loginButton);

buttonPanel.add(resetButton);

// Add components to the frame

frame.add(usernameLabel);

frame.add(usernameField);

frame.add(passwordLabel);

frame.add(passwordField);

frame.add(new JLabel()); // Empty space

frame.add(buttonPanel); // Add the button panel with the buttons side by side

frame.add(messageLabel); // To display the message

// Handle the login button click event

loginButton.addActionListener (new ActionListener(){

public void actionPerformed(ActionEvent e) {

String username = usernameField.getText();

char[] password = passwordField.getPassword();

String passwordStr = new String(password);

// Simple validation for login

if (username.equals("admin") && passwordStr.equals("password123"))

{

messageLabel.setText("Login successful!");

messageLabel.setForeground(Color.GREEN);

} else

{

messageLabel.setText("Invalid username or password.");

messageLabel.setForeground(Color.RED);

}

}

});

resetButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

usernameField.setText("");

passwordField.setText("");

messageLabel.setText("");

}

});

frame.setLocationRelativeTo(null);

frame.setVisible(true);

}

}

**Q.2 Write a program to display a string “Hello World” in a frame window with red color as background.**

import javax.swing.\*;

import java.awt.\*;

public class HelloWorldFrame

{

public static void main(String[] args)

{

JFrame frame = new JFrame("Hello World Frame");

frame.setSize(400, 200);

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

frame.getContentPane().setBackground(Color.RED);

JLabel label = new JLabel("Hello World", JLabel.CENTER);

label.setForeground(Color.WHITE);

frame.add(label);

frame.setLocationRelativeTo(null);

frame.setVisible(true);

}

}

**Q.3 Write a java program to design registration form. ( Use maximum Swing component in form).**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class RegistrationForm {

public static void main(String[] args) {

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

frame.setSize(400, 400);

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

frame.setLayout(new GridLayout(7, 2, 10, 10));

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

JTextField nameField = new JTextField(20);

JLabel usernameLabel = new JLabel("Username:");

JTextField usernameField = new JTextField(20);

JLabel passwordLabel = new JLabel("Password:");

JPasswordField passwordField = new JPasswordField(20);

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

JRadioButton maleRadio = new JRadioButton("Male");

JRadioButton femaleRadio = new JRadioButton("Female");

ButtonGroup genderGroup = new ButtonGroup();

genderGroup.add(maleRadio);

genderGroup.add(femaleRadio);

JLabel hobbiesLabel = new JLabel("Hobbies:");

JCheckBox readingCheck = new JCheckBox("Reading");

JCheckBox sportsCheck = new JCheckBox("Sports");

JButton registerButton = new JButton("Register");

JButton resetButton = new JButton("Reset");

frame.add(nameLabel);

frame.add(nameField);

frame.add(usernameLabel);

frame.add(usernameField);

frame.add(passwordLabel);

frame.add(passwordField);

frame.add(genderLabel);

JPanel genderPanel = new JPanel();

genderPanel.add(maleRadio);

genderPanel.add(femaleRadio);

frame.add(genderPanel);

frame.add(hobbiesLabel);

JPanel hobbiesPanel = new JPanel();

hobbiesPanel.add(readingCheck);

hobbiesPanel.add(sportsCheck);

frame.add(hobbiesPanel);

frame.add(registerButton);

frame.add(resetButton);

registerButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

String name = nameField.getText();

String username = usernameField.getText();

char[] password = passwordField.getPassword();

String gender = maleRadio.isSelected() ? "Male" : (femaleRadio.isSelected() ? "Female" : "Not selected");

String hobbies = "";

if (readingCheck.isSelected()) hobbies += "Reading ";

if (sportsCheck.isSelected()) hobbies += "Sports ";

if (name.isEmpty() || username.isEmpty() || password.length == 0) {

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

} else {

JOptionPane.showMessageDialog(frame, "Registration Successful!\nName: " + name + "\nUsername: " + username + "\nGender: " + gender + "\nHobbies: " + hobbies, "Success", JOptionPane.INFORMATION\_MESSAGE);

}

}

});

resetButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

nameField.setText("");

usernameField.setText("");

passwordField.setText("");

genderGroup.clearSelection();

readingCheck.setSelected(false);

sportsCheck.setSelected(false);

}

});

frame.setLocationRelativeTo(null);

frame.setVisible(true);

}

}

**Q.4.Write a java program to design email registration form.( Use maximum Swing component in form).**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class EmailRegistrationForm {

public static void main(String[] args) {

JFrame frame = new JFrame("Email Registration Form");

frame.setLayout(new GridLayout(7, 2, 10, 10));

frame.setSize(400, 300);

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

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

JTextField nameField = new JTextField(20);

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

JTextField emailField = new JTextField(20);

JLabel usernameLabel = new JLabel("Username:");

JTextField usernameField = new JTextField(20);

JLabel passwordLabel = new JLabel("Password:");

JPasswordField passwordField = new JPasswordField(20);

JLabel confirmPasswordLabel = new JLabel("Confirm Password:");

JPasswordField confirmPasswordField = new JPasswordField(20);

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

JRadioButton maleRadio = new JRadioButton("Male");

JRadioButton femaleRadio = new JRadioButton("Female");

ButtonGroup genderGroup = new ButtonGroup();

genderGroup.add(maleRadio);

genderGroup.add(femaleRadio);

JCheckBox termsCheckBox = new JCheckBox("I agree to the terms and conditions");

JButton submitButton = new JButton("Submit");

JButton resetButton = new JButton("Reset");

frame.add(nameLabel);

frame.add(nameField);

frame.add(emailLabel);

frame.add(emailField);

frame.add(usernameLabel);

frame.add(usernameField);

frame.add(passwordLabel);

frame.add(passwordField);

frame.add(confirmPasswordLabel);

frame.add(confirmPasswordField);

frame.add(genderLabel);

JPanel genderPanel = new JPanel();

genderPanel.add(maleRadio);

genderPanel.add(femaleRadio);

frame.add(genderPanel);

frame.add(termsCheckBox);

JPanel buttonPanel = new JPanel();

buttonPanel.add(submitButton);

buttonPanel.add(resetButton);

frame.add(buttonPanel);

submitButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

String fullName = nameField.getText();

String email = emailField.getText();

String username = usernameField.getText();

char[] password = passwordField.getPassword();

char[] confirmPassword = confirmPasswordField.getPassword();

String gender = maleRadio.isSelected() ? "Male" : (femaleRadio.isSelected() ? "Female" : "Not selected");

boolean termsAccepted = termsCheckBox.isSelected();

if (fullName.isEmpty() || email.isEmpty() || username.isEmpty() || password.length == 0 || !String.valueOf(password).equals(String.valueOf(confirmPassword))) {

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

} else if (!termsAccepted) {

JOptionPane.showMessageDialog(frame, "You must agree to the terms and conditions.", "Error", JOptionPane.WARNING\_MESSAGE);

} else {

JOptionPane.showMessageDialog(frame, "Registration Successful!\nFull Name: " + fullName + "\nUsername: " + username + "\nGender: " + gender, "Success", JOptionPane.INFORMATION\_MESSAGE);

}

}

});

resetButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

nameField.setText("");

emailField.setText("");

usernameField.setText("");

passwordField.setText("");

confirmPasswordField.setText("");

genderGroup.clearSelection();

termsCheckBox.setSelected(false);

}

});

frame.setLocationRelativeTo(null);

frame.setVisible(true);

}

}

**SET C**

1. **Write a program to design and implement the following GUI.**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class MenuExample {

public static void main(String[] args)

{

JFrame frame = new JFrame("Menu Example");

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

JMenuBar menuBar = new JMenuBar();

JMenu fileMenu = new JMenu("File");

JMenuItem newItem = new JMenuItem("New");

JMenuItem openItem = new JMenuItem("Open");

JMenuItem saveItem = new JMenuItem("Save");

JMenuItem exitItem = new JMenuItem("Exit");

fileMenu.add(newItem);

fileMenu.add(openItem);

fileMenu.add(saveItem);

fileMenu.addSeparator();

fileMenu.add(exitItem);

JMenu editMenu = new JMenu("Edit");

JMenuItem cutItem = new JMenuItem("Cut");

JMenuItem copyItem = new JMenuItem("Copy");

JMenuItem pasteItem = new JMenuItem("Paste");

JMenuItem selectAllItem = new JMenuItem("Select All");

editMenu.add(cutItem);

editMenu.add(copyItem);

editMenu.add(pasteItem);

editMenu.add(selectAllItem);

JMenu helpMenu = new JMenu("Help");

JMenuItem aboutItem = new JMenuItem("About");

helpMenu.add(aboutItem);

menuBar.add(fileMenu);

menuBar.add(editMenu);

menuBar.add(helpMenu);

frame.setJMenuBar(menuBar);

JTextArea textArea = new JTextArea(20, 40);

frame.add(new JScrollPane(textArea), BorderLayout.CENTER);

newItem.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

textArea.setText("");

}

});

openItem.addActionListener(new ActionListener()

{

public void actionPerformed(ActionEvent e)

{

JOptionPane.showMessageDialog(frame, "Open File functionality", "Open", JOptionPane.INFORMATION\_MESSAGE);

}

});

saveItem.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

JOptionPane.showMessageDialog(frame, "Save File functionality", "Save", JOptionPane.INFORMATION\_MESSAGE);

}

});

exitItem.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

System.exit(0);

}

});

cutItem.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

textArea.cut();

}

});

copyItem.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

textArea.copy();

}

});

pasteItem.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

textArea.paste();

}

});

selectAllItem.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

textArea.selectAll();

}

});

aboutItem.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

JOptionPane.showMessageDialog(frame, "Menu Example\nVersion 1.0", "About", JOptionPane.INFORMATION\_MESSAGE);

}

});

frame.setSize(500, 400);

frame.setLocationRelativeTo(null);

frame.setVisible(true);

}

}

**Q.2 Write a java Program to change the color of frame using Swing components.**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class BasicColor

{

public static void main(String[] args)

{

JFrame frame = new JFrame("Color Change Frame");

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

frame.setSize(400, 200);

frame.setLayout(new FlowLayout());

JButton redButton = createButton("Red", Color.RED);

JButton greenButton = createButton("Green", Color.GREEN);

frame.add(redButton);

frame.add(greenButton);

frame.setVisible(true);

}

private static JButton createButton(String text, final Color color) {

JButton button = new JButton(text);

button.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

JFrame frame = (JFrame) SwingUtilities.getWindowAncestor((Component) e.getSource());

frame.getContentPane().setBackground(color);

}

});

return button;

}

}

**Q.3 Write a program to design following GUI using JTextArea. Write a code to display the number 71 of words and characters of text in JLabel. Use JScrollPane to get scrollbars for JTextArea**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class WordCharacterCounterWithScroll {

public static void main(String[] args) {

// Create the frame

JFrame frame = new JFrame("Word and Character Counter");

// Set the size of the window

frame.setSize(400, 300);

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

// Create a JTextArea for text input with line wrap and word wrap

JTextArea textArea = new JTextArea(5, 30);

textArea.setLineWrap(true);

textArea.setWrapStyleWord(true);

// Add a JScrollPane to provide scrollbars

JScrollPane scrollPane = new JScrollPane(textArea);

frame.add(scrollPane, BorderLayout.CENTER);

// Create labels to display the word and character count

JLabel wordLabel = new JLabel("Words: 0");

JLabel charLabel = new JLabel("Characters: 0");

// Create a button to count words and characters

JButton countButton = new JButton("Count Words");

// Add action listener to the button

countButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

String text = textArea.getText(); // Get the text from the JTextArea

int wordCount = text.split("\\s+").length; // Count words by splitting on spaces

int charCount = text.length(); // Count characters

// Update the labels with the word and character count

wordLabel.setText("Words: " + wordCount);

charLabel.setText("Characters: " + charCount);

}

});

// Create a panel for the labels and button

JPanel panel = new JPanel();

panel.add(wordLabel);

panel.add(charLabel);

panel.add(countButton);

// Add the panel to the frame

frame.add(panel, BorderLayout.SOUTH);

// Make the frame visible

frame.setVisible(true);

}

}

**Q.4 Write a Program to design the following GUI by using swing component JComboBox. On click of the show button display the selected language on JLabel.**

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class ColorSelectorBasic {

public static void main(String[] args) {

// Create a frame

JFrame frame = new JFrame("Color Selector");

// Set the size of the window

frame.setSize(300, 150);

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

frame.setLayout(new FlowLayout());

// Create a JComboBox with color options

String[] colors = {"Red", "Green", "Blue"};

JComboBox<String> colorComboBox = new JComboBox<>(colors);

// Create a JButton to show the selected color

JButton showButton = new JButton("Show");

// Create a JLabel to display the selected color

JLabel label = new JLabel("Selected Color: None");

// Add action listener to the button

showButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

String selectedColor = (String) colorComboBox.getSelectedItem();

label.setText("Selected Color: " + selectedColor);

}

});

// Add the components to the frame

frame.add(colorComboBox);

frame.add(showButton);

frame.add(label);

// Set the frame visible

frame.setVisible(true);

}

}