**SIR SYED C@SE INSTITUTE OF TECHNOLOGY ISLAMABAD**



**PROJECT NAME**

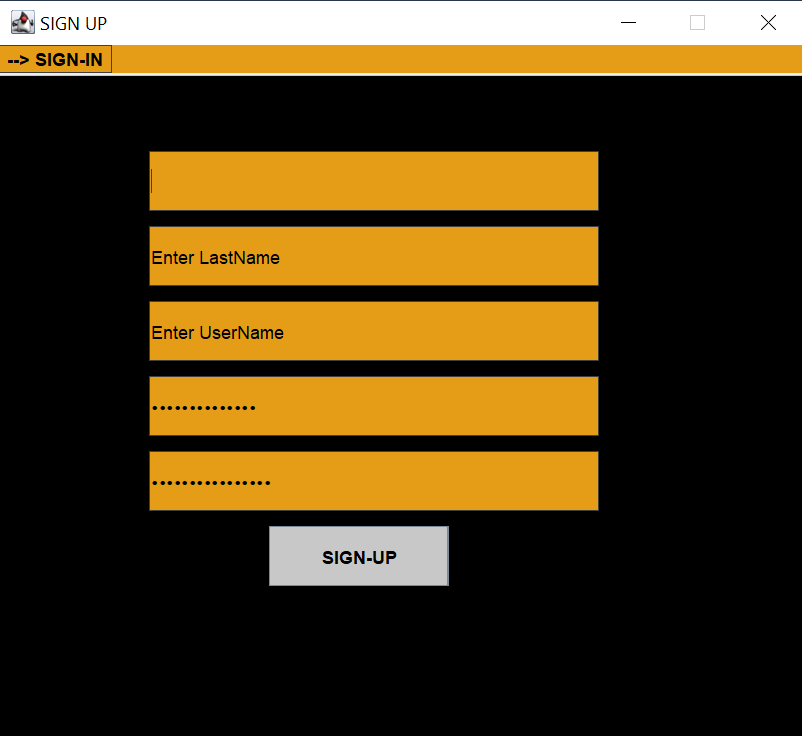
**YUM YUM CAMPUS BITES FOOD APPLICATION**

**(SIGN UP FORM)**

**CODE:**

package FOOD\_PANDA\_APP;  
import javax.swing.\*;  
import java.awt.\*;  
import java.awt.event.FocusEvent;  
import java.awt.event.FocusListener;  
import java.io.\*;  
  
public class SignUp2  
{  
 static final String *FILE\_PATH* = "user\_records.txt";  
// static final String FILE\_PATH = "user\_records.txt";  
 // csv files are supported by ultimate version so this version can't support csv file.  
 void SignUp2()  
 {  
 JFrame frame = new JFrame("SIGN UP");  
 JMenuBar menuBar=new JMenuBar();  
 frame.setSize(550, 500);  
  
 JMenu SignIn;  
 SignIn = new JMenu("--> SIGN-IN");  
 Color Color3 = new Color(229, 156, 23); // RGB values for b  
 SignIn.setBackground(Color3);  
 SignIn.setForeground(Color.*black*);  
 SignIn.setBorder(BorderFactory.*createLineBorder*(Color.*darkGray*));  
 menuBar.add(SignIn);  
 JMenuItem SignIn1,Home1;  
  
 SignIn1=new JMenuItem("--> SIGN-IN");  
 SignIn.add(SignIn1);  
 Color Color2 = new Color(229, 156, 23); // RGB values for b  
 SignIn1.setBackground(Color2);  
 SignIn1.setForeground(Color.*black*);  
 SignIn1.setBorder(BorderFactory.*createLineBorder*(Color.*darkGray*));  
 SignIn1.addActionListener(e -> {  
 SignIn2 s2=new SignIn2();  
 frame.dispose();  
 s2.SignIn2();  
 });  
  
 Home1=new JMenuItem("<-- HOME");  
 SignIn.add(Home1);  
 Home1.setBorder(BorderFactory.*createLineBorder* (Color.*darkGray*));  
 Home1.setForeground(Color.*black*);  
 Color Color11 = new Color(229, 156, 23); // RGB values for b // rgb stands for red green blue  
 Home1.setBackground(Color11);  
  
 Home1.addActionListener(e -> {  
 Home h=new Home();  
 frame.dispose();  
 h.Home1();  
 });  
  
 frame.setJMenuBar(menuBar);  
 Color Color1 = new Color(229, 156, 23); // RGB values for b  
 menuBar.setBackground(Color1);  
  
 // Create panels and components  
 JPanel panel1 = *createTextFieldPanel*("Enter FirstName");  
 JPanel panel2 = *createTextFieldPanel*("Enter LastName");  
 JPanel panel3 = *createTextFieldPanel*("Enter UserName");  
 JPanel panel4 = *createPasswordFieldPanel*("Enter Password");  
 JPanel panel5 = *createPasswordFieldPanel*("Confirm Password");  
 JPanel panel = *createButtonPanel*();  
  
 // Set layouts for panels  
 frame.setLayout(null); // Use null layout for manual component placement  
 *setPanelBounds*(panel1, 100, 50, 300, 40);  
 *setPanelBounds*(panel2, 100, 100, 300, 40);  
 *setPanelBounds*(panel3, 100, 150, 300, 40);  
 *setPanelBounds*(panel4, 100, 200, 300, 40);  
 *setPanelBounds*(panel5, 100, 250, 300, 40);  
 *setPanelBounds*(panel, 180, 300, 120, 40);  
  
 // Add panels to the frame  
 frame.add(panel1);  
 frame.add(panel2);  
 frame.add(panel3);  
 frame.add(panel4);  
 frame.add(panel5);  
 frame.add(panel);  
  
 // Button action  
 JButton button1 = (JButton) panel.getComponent(0);  
 button1.addActionListener(e -> {  
 SignIn2 s2=new SignIn2();  
 frame.dispose();  
 s2.SignIn2();  
 saveUserRecord(panel2, panel3, panel4);  
 JOptionPane.*showMessageDialog*(frame, "You have registered your account!");  
 System.*out*.println("Signing up...");  
 });  
  
 frame.getContentPane().setBackground(Color.*black*);  
 frame.setDefaultCloseOperation(JFrame.*DISPOSE\_ON\_CLOSE*);  
 frame.setResizable(false);  
 frame.setVisible(true);  
  
 }  
  
 private static JPanel createTextFieldPanel(String placeholder) {  
 JPanel panel = new JPanel();  
 JTextField textField = new JTextField(placeholder);  
 textField.setForeground(Color.*black*);  
 textField.setCaretColor(Color.*darkGray*);// it is used for changing color of cursor  
 textField.setBorder(BorderFactory.*createLineBorder*(Color.*darkGray*));  
  
 Color Color4 = new Color(229, 156, 23); // RGB values for b  
 textField.setBackground(Color4);  
  
  
 // Add a focus listener to handle placeholder behavior  
 textField.addFocusListener(new FocusListener() {  
 @Override  
 public void focusGained(FocusEvent e) {  
 if (textField.getText().equals(placeholder)) {  
 textField.setText(""); // Clear the placeholder text when focused  
 textField.setForeground(Color.*black*); // Change text color when typing  
 }  
 }  
  
 @Override  
 public void focusLost(FocusEvent e) {  
 if (textField.getText().isEmpty()) {  
 textField.setForeground(Color.*black*);  
 textField.setText(placeholder); // Reset placeholder if no input provided  
 }  
 }  
 });  
  
 panel.add(textField);  
 return panel;  
 }  
  
  
 private static JPanel createPasswordFieldPanel(String placeholder) {  
 JPanel panel = new JPanel();  
 JPasswordField passwordField = new JPasswordField(placeholder);  
 passwordField.setForeground(Color.*black*);  
 passwordField.setBorder(BorderFactory.*createLineBorder*(Color.*darkGray*));  
 Color Color5 = new Color(229, 156, 23); // RGB values for b  
 passwordField.setBackground(Color5);  
  
  
 // Add a focus listener to handle placeholder behavior  
 passwordField.addFocusListener(new FocusListener() {  
 @Override  
 public void focusGained(FocusEvent e) {  
 char[] pass = passwordField.getPassword();  
 String password = new String(pass);  
 if (password.equals(placeholder)) {  
 passwordField.setText(""); // Clear the placeholder text when focused  
 passwordField.setForeground(Color.*black*); // Change text color when typing  
 }  
 }  
  
 @Override  
 public void focusLost(FocusEvent e) {  
 char[] pass = passwordField.getPassword();  
 String password = new String(pass);  
 if (password.isEmpty()) {  
 passwordField.setForeground(Color.*black*);  
 passwordField.setText(placeholder); // Reset placeholder if no input provided  
 }  
 }  
 });  
  
 panel.add(passwordField);  
 return panel;  
 }  
  
  
 private static JPanel createButtonPanel() {  
 JPanel panel = new JPanel();  
 JButton button = new JButton("SIGN-UP");  
 Color Color5 = new Color(200, 200, 200); // RGB values for b  
 button.setBackground(Color5);  
 button.setForeground(Color.*black*);  
 panel.add(button);  
 return panel;  
 }  
  
 private static void setPanelBounds(JPanel panel, int x, int y, int width, int height) {  
 panel.setLayout(new GridLayout(1, 1));  
 panel.setBounds(x, y, width, height);  
 }  
  
 private void saveUserRecord(JPanel panelLastName, JPanel panelUserName, JPanel panelPassword) {  
 String lastName = getTextFieldInput(panelLastName);  
 String userName = getTextFieldInput(panelUserName);  
 String password = getPasswordFieldInput(panelPassword);  
  
 if (!lastName.isEmpty() && !userName.isEmpty() && !password.isEmpty()) {  
 try (BufferedWriter writer = new BufferedWriter(new FileWriter(*FILE\_PATH*, true))) {  
 writer.write(userName + "," + password + "," + lastName + "\n");  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 } else {  
 JOptionPane.*showMessageDialog*(null, "Please fill all fields.");  
 }  
 }  
  
 private String getTextFieldInput(JPanel panel) {  
 JTextField textField = (JTextField) panel.getComponent(0);  
 String input = textField.getText().equals(textField.getName()) ? "" : textField.getText();  
 return input;  
 }  
  
 private String getPasswordFieldInput(JPanel panel) {  
 JPasswordField passwordField = (JPasswordField) panel.getComponent(0);  
 char[] pass = passwordField.getPassword();  
 String password = new String(pass).equals(passwordField.getName()) ? "" : new String(pass);  
 return password;  
 }  
  
 public static boolean authenticateUser(String username, String password) {  
 try (BufferedReader br = new BufferedReader(new FileReader(*FILE\_PATH*))) {  
 String line;  
 while ((line = br.readLine()) != null) {  
 String[] parts = line.split(",");  
 if (parts.length >= 2 && parts[0].equals(username) && parts[1].equals(password)) {  
 return true;  
 }  
 }  
 } catch (IOException e) {  
 e.printStackTrace();  
 }  
 return false;  
 }  
}

**OUTPUT:**



**After Sign Up and filling following panels we get following panel and shift to Sign In Form.**

