Nama : Rafi Ikhsanul Hakim

NPM : 223040095

Kelas : C

Matkul : Praktikum Pemrograman II

Github Repo : <https://github.com/RafiAkem/PrakProII/blob/main/Tugas2/Tugas2.java>

1. Pengaturan Dasar JFrame
2. public class Tugas2 extends JFrame {
3. public Tugas2() {
4. this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

* Membuat kelas Tugas2 yang meng extend dari JFrame, lalu mengatur operasi default saat jendela ditutup agar aplikasi keluar

2. Menu Bar

JMenuBar menuBar = new JMenuBar();

JMenu menu = new JMenu("Menu");

JMenuItem resetMenuItem = new JMenuItem("Reset");

JMenuItem exitMenuItem = new JMenuItem("Exit");

menu.add(resetMenuItem);

menu.add(exitMenuItem);

menuBar.add(menu);

this.setJMenuBar(menuBar);

* Membuat menu bar dengan menu yang berisikan “Reset” dan “Exit” dimana jika reset diklik akan mengosongkan semua input, “Exit” untuk keluar dari aplikasi

3. Label dan TextField

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

        JTextField nameField = new JTextField();

        JLabel phoneLabel = new JLabel("No. Telepon: ");

        JTextField phoneField = new JTextField();

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

        JTextField addressField = new JTextField();

* Membuat Label dan text field untuk input nama. Nomor telepon, dan Alamat

4. JList untuk Jenis Tabungan

        JLabel savingsLabel = new JLabel("Jenis Tabungan: ");

        String[] savingsTypes = {"Tabungan A", "Tabungan B", "Tabungan C", "Tabungan D"};

        JList<String> savingsList = new JList<>(savingsTypes);

        savingsList.setVisibleRowCount(2);

        savingsList.setLayoutOrientation(JList.HORIZONTAL\_WRAP);

        JScrollPane savingsScrollPane = new JScrollPane(savingsList);

* Membuat label dan daftar pilihan (Jlist) untuk jenis Tabungan dengan beberapa opsi.

1. JSlider untuk Frekuensi Transaksi
2. JLabel transactionLabel = new JLabel("Frekuensi Transaksi per Bulan: ");
3. JSlider transactionFrequencySlider = new JSlider(1, 100, 1);
4. transactionFrequencySlider.setPaintLabels(true);
5. transactionFrequencySlider.setPaintTicks(true);
6. transactionFrequencySlider.setMajorTickSpacing(9);
7. JLabel transactionValueLabel = new JLabel("");
8. transactionFrequencySlider.addChangeListener(e -> {
9. transactionValueLabel.setText(String.valueOf(transactionFrequencySlider.getValue()));
10. });

* Membuat label dan slider untuk memilih frekuensi transaksi per bulan.
* Menampilkan nilai slider secara dinamis.

6. JPasswordField untuk Password dan konfirmasi Password

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

        JPasswordField passwordField = new JPasswordField();

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

        JPasswordField confirmPasswordField = new JPasswordField();

* Membuat label dan field password untuk input password dan konfirmasi password.

7. JSpinner untuk Tanggal Lahir

        JLabel dobLabel = new JLabel("Tanggal Lahir: ");

        JSpinner dobSpinner = new JSpinner(new SpinnerDateModel());

        JSpinner.DateEditor dateEditor = new JSpinner.DateEditor(dobSpinner, "dd/MM/yyyy");

        dobSpinner.setEditor(dateEditor);

* Membuat label dan spiner untuk input tanggal lahir dengan format “dd/mm/yyyy”.

8. Tombol Simpan dan TextArea untuk Output

        JButton button = new JButton("Simpan");

        JTextArea textOutput = new JTextArea(5, 20);

        textOutput.setEditable(false);

        JScrollPane textOutputScrollPane = new JScrollPane(textOutput);

        button.addActionListener(new ActionListener() {

            public void actionPerformed(ActionEvent e) {

                String name = nameField.getText();

                String phone = phoneField.getText();

                String address = addressField.getText();

                String savingsType = savingsList.getSelectedValue();

                int transactionFrequency = transactionFrequencySlider.getValue();

                String password = new String(passwordField.getPassword());

                String confirmPassword = new String(confirmPasswordField.getPassword());

                Date dob = (Date) dobSpinner.getValue();

                textOutput.setText("Nama: " + name + "\n");

                textOutput.append("No. Telepon: " + phone + "\n");

                textOutput.append("Alamat: " + address + "\n");

                textOutput.append("Jenis Tabungan: " + savingsType + "\n");

                textOutput.append("Frekuensi Transaksi per Bulan: " + transactionFrequency + "\n");

                textOutput.append("Tanggal Lahir: " + dateEditor.getFormat().format(dob) + "\n");

                textOutput.append("Password Match: " + password.equals(confirmPassword) + "\n");

                nameField.setText("");

                phoneField.setText("");

                addressField.setText("");

                savingsList.clearSelection();

                transactionFrequencySlider.setValue(1);

                passwordField.setText("");

                confirmPasswordField.setText("");

                dobSpinner.setValue(new Date());

            }

        });

* Membuat tombol “Simpan” dan area teks untuk menampilkan output.
* Saat tombol “Simpan” diklik, data dari input diambil dan ditampilkan di area teks, kemudian input direset.

9. Function untuk Menu Reset dan Exit

        resetMenuItem.addActionListener(new ActionListener() {

            public void actionPerformed(ActionEvent e) {

                nameField.setText("");

                phoneField.setText("");

                addressField.setText("");

                savingsList.clearSelection();

                transactionFrequencySlider.setValue(1);

                passwordField.setText("");

                confirmPasswordField.setText("");

                dobSpinner.setValue(new Date());

                textOutput.setText("");

            }

        });

        exitMenuItem.addActionListener(new ActionListener() {

            public void actionPerformed(ActionEvent e) {

                System.exit(0);

            }

        });

* Memberikan fungsi untuk menu Reset dan Exit

10. Layout Komponen

        GroupLayout layout = new GroupLayout(getContentPane());

        getContentPane().setLayout(layout);

        layout.setAutoCreateGaps(true);

        layout.setAutoCreateContainerGaps(true);

        layout.setHorizontalGroup(layout.createParallelGroup(GroupLayout.Alignment.LEADING)

            .addComponent(nameLabel)

            .addComponent(nameField)

            .addComponent(phoneLabel)

            .addComponent(phoneField)

            .addComponent(addressLabel)

            .addComponent(addressField)

            .addComponent(savingsLabel)

            .addComponent(savingsScrollPane)

            .addComponent(transactionLabel)

            .addGroup(layout.createSequentialGroup()

                .addComponent(transactionFrequencySlider)

                .addComponent(transactionValueLabel))

            .addComponent(passwordLabel)

            .addComponent(passwordField)

            .addComponent(confirmPasswordLabel)

            .addComponent(confirmPasswordField)

            .addComponent(dobLabel)

            .addComponent(dobSpinner)

            .addComponent(button)

            .addComponent(textOutputScrollPane)

        );

        layout.setVerticalGroup(layout.createSequentialGroup()

            .addComponent(nameLabel)

            .addComponent(nameField)

            .addComponent(phoneLabel)

            .addComponent(phoneField)

            .addComponent(addressLabel)

            .addComponent(addressField)

            .addComponent(savingsLabel)

            .addComponent(savingsScrollPane)

            .addComponent(transactionLabel)

            .addGroup(layout.createParallelGroup(GroupLayout.Alignment.BASELINE)

                .addComponent(transactionFrequencySlider)

                .addComponent(transactionValueLabel))

            .addComponent(passwordLabel)

            .addComponent(passwordField)

            .addComponent(confirmPasswordLabel)

            .addComponent(confirmPasswordField)

            .addComponent(dobLabel)

            .addComponent(dobSpinner)

            .addComponent(button)

            .addComponent(textOutputScrollPane)

        );

        this.setSize(500, 800);

    }

* Mengatur layout komponen menggunakan GroupLayout untuk mengatur posisi dan ukuran komponen dalam frame.

11. Metode Main

    public static void main(String[] args) {

        javax.swing.SwingUtilities.invokeLater(new Runnable() {

            public void run() {

                Tugas2 frame = new Tugas2();

                frame.setVisible(true);

            }

        });

    }

}

* Main method untuk menjalankan aplikasi.
* Menggunakan SwingUtilities.invokeLater untuk memastikan GUI dibuat dan diperbarui di Event Dispatch Thread

OUTPUT :



