

LAPORAN PRAKTIKUM
BAHASA PEMROGRAMAN 2
MODUL 3

Dosen pengampu : Yulyanto, S.Kom., M.TI.



Disusun oleh :

Nama : Muhammad Rizal Nurfirdaus
NIM : 20230810088
Jadwal : Rabu, 14:40 – 16:15
Kelas : TINFC-2023-04

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS KUNINGAN

2025

DAFTAR ISI

DAFTAR ISI	ii
BAB I	1
PRETEST	1
BAB II	2
PRAKTIKUM	2
BAB III	6
POSTTEST	6
BAB IV	9
TUGAS	9
BAB V	13
KESIMPULAN	13

BAB I

PRETEST

1. Tuliskan Konstruktor dari kelas JLabel dan JButton
Kerjakan pada selembar kertas dengan waktu 5 menit dan dikumpulkan melalui Asisten lab. import javax.swing.JOptionPane;

Konstruktor JLabel :

```
JLabel label1 = new JLabel();           // Konstruktor tanpa parameter
JLabel label2 = new JLabel("Teks Label"); // Konstruktor dengan teks
JLabel label3 = new JLabel(new ImageIcon("gambar.png")); // Konstruktor
dengan ikon
JLabel label4 = new JLabel("Teks", JLabel.CENTER); // Teks dan alignment
```

Konstruktor JButton :

```
JButton button1 = new JButton();           // Konstruktor tanpa parameter
JButton button2 = new JButton("Teks Tombol"); // Konstruktor dengan teks
JButton button3 = new JButton(new ImageIcon("ikon.png")); // Konstruktor
dengan ikon
JButton button4 = new JButton("Teks", new ImageIcon("ikon.png")); // Teks dan
ikon
```

BAB II

PRAKTIKUM

1. Buatlah User Interface dengan Menggunakan komponen SWING dengan tampilan sebagai berikut :



Lalu memasukkan kode sebagai berikut :

```
public FRegister() {  
    initComponents();  
}  
  
private void simpan() {  
    String nama, jk, prodi, alamat;  
    nama = jTextField1.getText(); // Assuming jTextField1 is the input for name  
  
    // Penanganan Jenis Kelamin  
    jk = "";  
    if (jRadioButton1.isSelected()) {  
        jk = jRadioButton1.getText(); // Assuming jRadioButton1 is for "PRIA"  
    } else if (jRadioButton2.isSelected()) {
```

```

        jk = jRadioButton2.getText(); // Assuming jRadioButton2 is for
"WANITA"

    }

    // Penanganan Prodi

    prodi = (String) jComboBox1.getSelectedItem(); // Assuming jComboBox1
is the dropdown for Prodi

    // Penanganan Hobi

    StringBuilder hobi = new StringBuilder();

    if (jCheckBox1.isSelected()) {

        hobi.append(jCheckBox1.getText()).append(","); // Assuming jCheckBox1
is for "MEMBACA"

    }

    if (jCheckBox2.isSelected()) {

        hobi.append(jCheckBox2.getText()).append(","); // Assuming jCheckBox2
is for "NONTON"

    }

    if (jCheckBox3.isSelected()) {

        hobi.append(jCheckBox3.getText()).append(","); // Assuming jCheckBox3
is for "OLAHRAGA"

    }

    if (jCheckBox4.isSelected()) {

        hobi.append(jCheckBox4.getText()).append(","); // Assuming jCheckBox4
is for "SHOPING"

    }

    // Penanganan Alamat

    alamat = jTextArea1.getText(); // Assuming jTextArea1 is the input for
address

    // Output Hasil

```

```

jTextArea2.setText("DATA REGISTER \n"
    + "1. Nama : " + nama + "\n"
    + "2. Jenis Kelamin : " + jk + "\n"
    + "3. Prodi : " + prodi + "\n"
    + "4. Hobi : " + hobi + "\n"
    + "5. Alamat : " + alamat + "\n");
}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    simpan();
}

```

FORM REGISTER

NAMA:

JENIS KELAMIN: ☒ PRIA ☐ WANITA

PRODI:

HOBİ: ☒ MEMBACA ☐ NONTON
☒ OLAHRAGA ☐ SHOPING

ALAMAT:

REGISTER

DATA REGISTER
 1. Nama : Muhammad Rizal Nurfirdaus
 2. Jenis Kelamin : PRIA
 3. Prodi : TEKNIK INFORMATIKA S1
 4. Hobi : MEMBACA,OLAHRAGA,
 5. Alamat : Desa Sidaraja

Analisis : Kode di atas merupakan bagian dari program Java berbasis GUI (kemungkinan menggunakan Swing) yang menangani proses pendaftaran data pengguna. Method `simpan()` bertugas untuk mengambil input dari berbagai komponen GUI seperti `TextField`, `RadioButton`, `ComboBox`, `CheckBox`, dan `TextArea`. Input tersebut meliputi nama, jenis kelamin, program studi, hobi, dan alamat. Setelah semua data dikumpulkan, informasi tersebut ditampilkan kembali dalam `TextArea2` dalam format yang terstruktur.

BAB III

POSTTEST

1. Lengkapi Kode program di atas, sehingga, ketika button register di klik, maka Form Register kembali Kosong.
2. Lengkapi juga dengan Confirm Dialog YesNoOption ketika mengklik button Register.

Script program :

1.

```
private void resetForm() {  
    jTextField1.setText(""); // Reset name field  
    jRadioButton1.setSelected(false); // Reset gender selection  
    jRadioButton2.setSelected(false);  
    jComboBox1.setSelectedIndex(0); // Reset program selection  
    jCheckBox1.setSelected(false); // Reset hobbies  
    jCheckBox2.setSelected(false);  
    jCheckBox3.setSelected(false);  
    jCheckBox4.setSelected(false);  
    jTextArea1.setText(""); // Reset address field  
}
```

2.

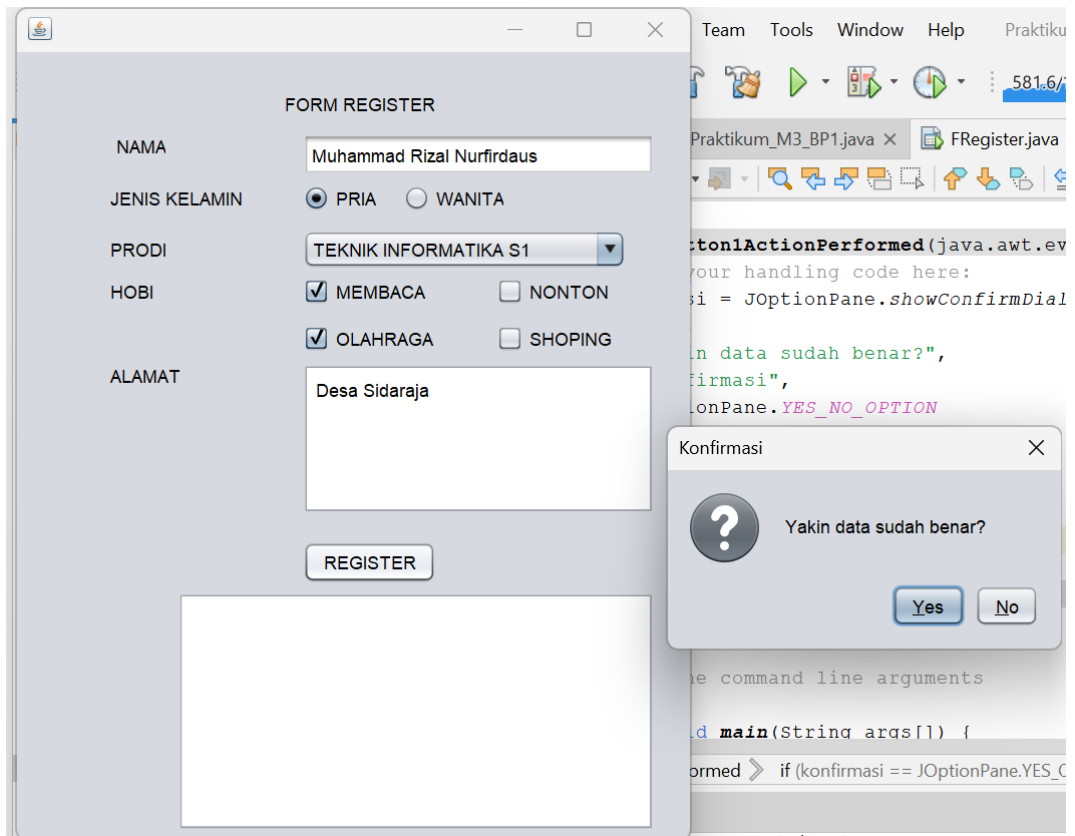
```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    int konfirmasi = JOptionPane.showConfirmDialog(  
        this,  
        "Yakin data sudah benar?",  
        "Konfirmasi",  
        JOptionPane.YES_NO_OPTION  
    );  
  
    if (konfirmasi == JOptionPane.YES_OPTION) {
```



```

        simpan();
        resetForm();
    }
}

```



FORM REGISTER

NAMA

JENIS KELAMIN ☐ PRIA ☐ WANITA

PRODI

HOBI ☐ MEMBACA ☐ NONTON
☐ OLAHRAGA ☐ SHOPING

ALAMAT

DATA REGISTER

1. Nama : Muhammad Rizal Nurfirdaus
2. Jenis Kelamin : PRIA
3. Prodi : TEKNIK INFORMATIKA S1
4. Hobi : MEMBACA, OLAHRAGA,
5. Alamat : Desa Sidaraja

Analisis : Penambahan kode `resetForm()` dan penggunaan `JOptionPane.showConfirmDialog` pada method `jButton1ActionPerformed` merupakan langkah yang tepat untuk meningkatkan interaksi dan pengalaman pengguna. Method `resetForm()` berfungsi untuk mengosongkan seluruh input form, sehingga pengguna dapat langsung mengisi data baru setelah menekan tombol *Register*. Ini membuat alur pengisian data menjadi lebih efisien dan bersih setelah satu kali pendaftaran berhasil dilakukan.

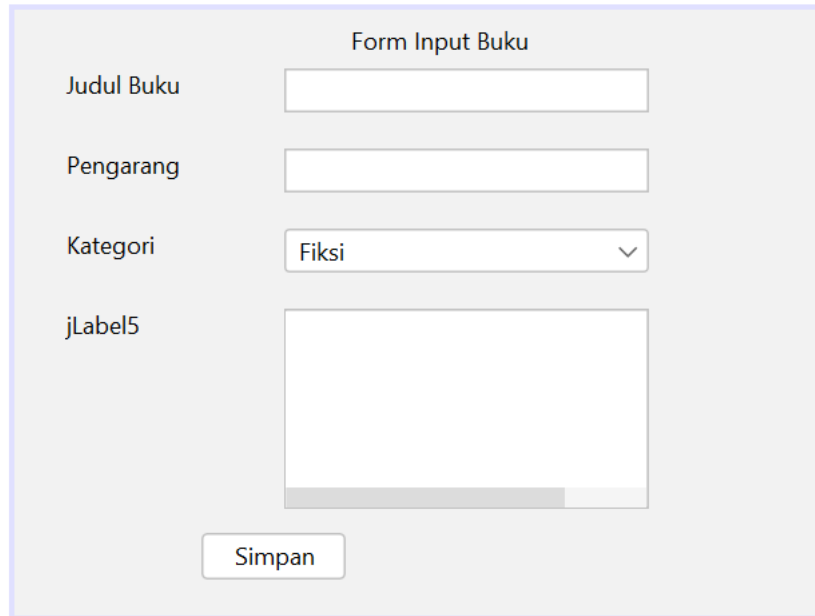
Sementara itu, penggunaan `JOptionPane` memberikan lapisan konfirmasi tambahan sebelum data disimpan, yang dapat mencegah kesalahan akibat klik tidak disengaja. Jika pengguna memilih "Yes", maka data akan disimpan melalui method `simpan()` dan form langsung dikosongkan. Jika memilih "No", maka tidak ada aksi yang dilakukan. Kombinasi fitur ini tidak hanya meningkatkan keamanan input data, tetapi juga memperkuat prinsip usability dalam antarmuka aplikasi.

BAB IV

TUGAS

1. Buatlah Form Input yang terdiri dari komponen swing jTextField, JComboBox, JTextArea, JOptionPane. (Studi Kasus Bebas).

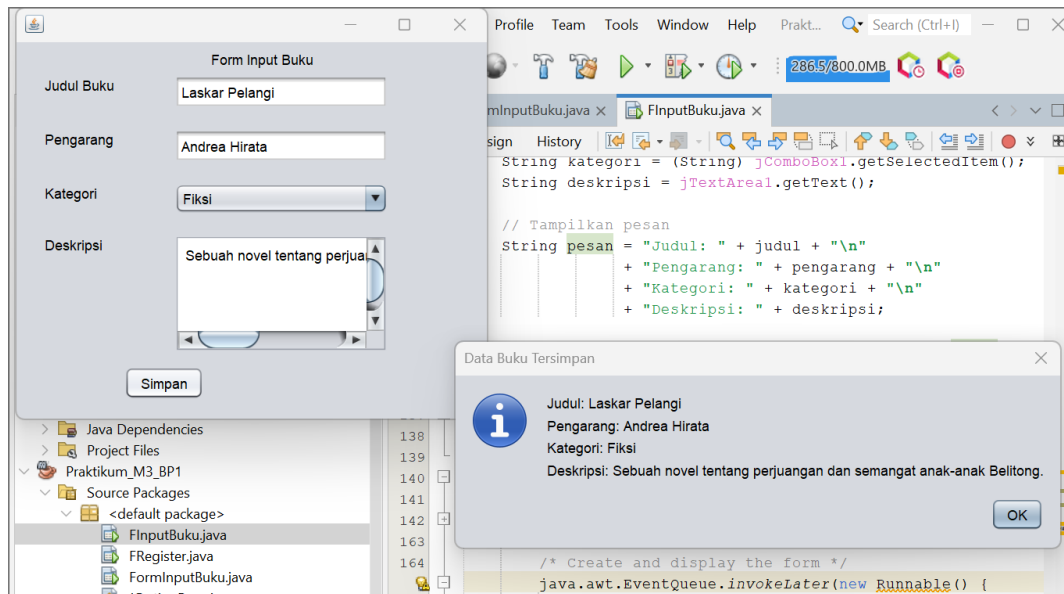
Tampilan :



The screenshot shows a Java Swing window titled "Form Input Buku". Inside the window, there are four input components arranged vertically: a text field for "Judul Buku", a text field for "Pengarang", a JComboBox for "Kategori" with "Fiksi" selected, and a JTextArea for "jLabel5". At the bottom center of the window is a button labeled "Simpan".

Script Code Di Tombol Simpan :

```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    String judul = jTextField1.getText();  
    String pengarang = jTextField2.getText();  
    String kategori = (String) jComboBox1.getSelectedItem();  
    String deskripsi = jTextArea1.getText();  
  
    // Tampilkan pesan  
    String pesan = "Judul: " + judul + "\n"  
        + "Pengarang: " + pengarang + "\n"  
        + "Kategori: " + kategori + "\n"  
        + "Deskripsi: " + deskripsi;  
  
    javax.swing.JOptionPane.showMessageDialog(null, pesan, "Data Buku  
Tersimpan", javax.swing.JOptionPane.INFORMATION_MESSAGE);  
}
```



Analisis : Program Form Input Buku ini merupakan aplikasi GUI sederhana berbasis Java Swing yang memungkinkan pengguna untuk menginput data buku ke dalam sistem. Form terdiri dari beberapa komponen seperti JTextField untuk memasukkan judul dan pengarang buku, JComboBox untuk memilih kategori buku, JTextArea untuk menuliskan deskripsi buku, serta sebuah tombol JButton untuk menyimpan data. Setelah tombol ditekan, data yang telah diisi akan ditampilkan kembali kepada pengguna melalui JOptionPane sebagai konfirmasi bahwa data berhasil disimpan. Aplikasi ini cocok digunakan sebagai simulasi awal dalam pengembangan sistem perpustakaan berbasis desktop.

2. Buatlah Aplikasi Perhitungan Konfersi Mata uang. Gunakan Komponen SWING.

Tampilan :

Script Program :

```
private void konversi() {
    try {
```

```

double nilai = Double.parseDouble(jTextField1.getText());
String pilihan = (String) jComboBox1.getSelectedItem();
double hasil;

switch (pilihan) {
    case "Rupiah ke Dollar":
        hasil = nilai / 15000;
        jTextField1.setText(String.format("Hasil: $ %.2f", hasil));
        break;
    case "Rupiah ke Yuan":
        hasil = nilai / 2100;
        jTextField1.setText(String.format("Hasil: ¥ %.2f", hasil));
        break;
    case "Dollar ke Rupiah":
        hasil = nilai * 15000;
        jTextField1.setText(String.format("Hasil: Rp %.2f", hasil));
        break;
    case "Yuan ke Rupiah":
        hasil = nilai * 2100;
        jTextField1.setText(String.format("Hasil: Rp %.2f", hasil));
        break;
    default:
        jTextField1.setText("Pilihan konversi tidak valid.");
}

} catch (NumberFormatException ex) {
    JOptionPane.showMessageDialog(this, "Masukkan angka yang valid!",
    "Error", JOptionPane.ERROR_MESSAGE);
}
}

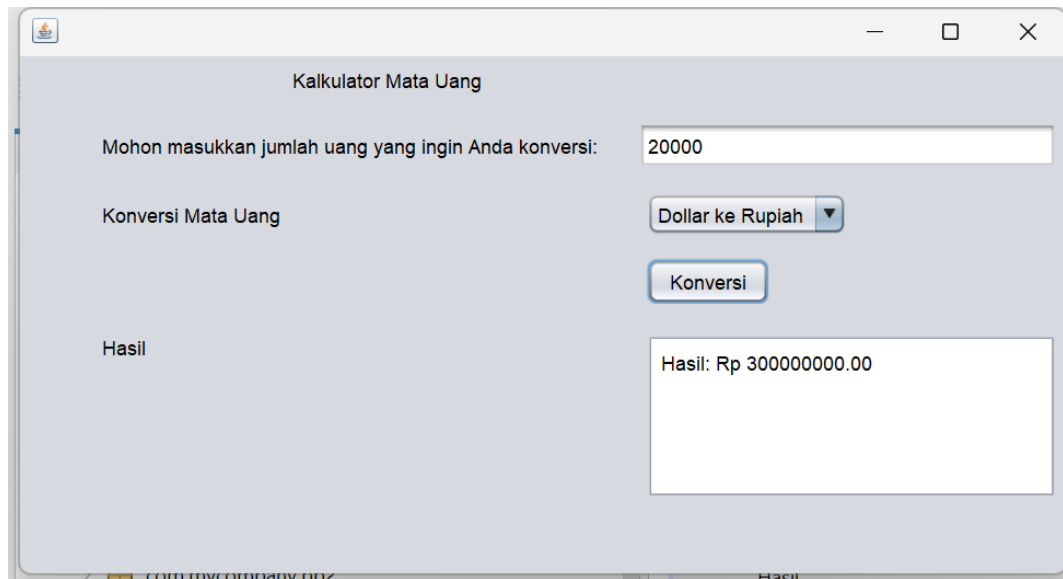
```

Masukkan di Tombol konversi script dibawah

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    konversi();
}

```



Analisis : Program **Konversi Mata Uang** di atas merupakan aplikasi Java berbasis GUI (Swing) yang memungkinkan pengguna mengonversi nilai mata uang dari satu jenis ke jenis lain, seperti Rupiah ke Dollar, Rupiah ke Yuan, dan sebaliknya. Fungsionalitas utama terletak pada metode konversi(), yang memanfaatkan switch-case untuk menentukan jenis konversi berdasarkan pilihan dari JComboBox1. Nilai input diambil dari jTextField1 dan hasil konversi ditampilkan dalam jTextArea1. Program ini juga menangani kesalahan input numerik menggunakan blok try-catch, yang meningkatkan ketahanan terhadap input yang tidak valid.

BAB V

KESIMPULAN

Selama proses praktikum ini, mahasiswa telah mempelajari bagaimana membangun antarmuka pengguna (user interface) menggunakan Java Swing. Pada bagian Pretest, mahasiswa diminta menuliskan beberapa bentuk konstruktor dari komponen JLabel dan JButton. Tujuannya agar mahasiswa mengenal cara dasar dalam membuat label dan tombol pada aplikasi GUI, baik dengan teks biasa, ikon, maupun gabungan dari keduanya. Ini menjadi pondasi awal sebelum masuk ke implementasi yang lebih kompleks di tahap praktikum.

Saat praktikum berlangsung, mahasiswa membangun sebuah form registrasi yang lebih lengkap. Di sini, mereka tidak hanya mengambil data dari berbagai komponen seperti JTextField, JRadioButton, JComboBox, JCheckBox, dan JTextArea, tapi juga belajar menyusun data tersebut agar tampil dengan rapi. Selain itu, ditambahkan juga fungsi konfirmasi menggunakan JOptionPane sebelum menyimpan data, serta pembersihan form setelah proses simpan. Ini melatih mahasiswa agar terbiasa membuat aplikasi yang tidak hanya berfungsi, tapi juga nyaman digunakan oleh pengguna.

Di bagian tugas, mahasiswa membuat dua aplikasi: form input data buku dan aplikasi konversi mata uang. Masing-masing tugas menggabungkan penggunaan berbagai komponen Swing dan penerapan logika pemrograman yang sesuai. Pada form input buku, data ditampilkan dalam kotak dialog sebagai konfirmasi penyimpanan. Sedangkan pada konversi mata uang, mahasiswa menerapkan logika perhitungan menggunakan switch dan juga menangani input yang tidak valid. Dengan selesainya semua bagian praktikum ini, mahasiswa diharapkan bisa membangun aplikasi desktop sederhana dengan tampilan yang interaktif dan fungsional sesuai kebutuhan pengguna.