

LAPORAN PRAKTIKUM  
ALGORITMA PEMROGRAMAN PEKAN DELAPAN  
*Graphical User Interface (GUI)*

disusun Oleh:

Zahira Nur Asyifa

2411532015

Dosen Pengampu: Dr. Wahyudi, S.T, M.T

Asisten Praktikum: Rahmad Dwirizki Olders



DEPARTEMEN INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI  
UNIVERSITAS ANDALAS  
2025

## **KATA PENGANTAR**

Puji syukur penulis panjatkan ke hadirat Allah SWT karena atas rahmat dan karunia-Nya, laporan praktikum Algoritma dan Pemrograman dengan topik perancangan dan pemrograman antarmuka grafis (GUI) menggunakan Java dapat diselesaikan dengan baik. Laporan ini disusun untuk memenuhi tugas praktikum pekan keenam mata kuliah praktikum Algoritma dan Pemrograman, sekaligus sebagai sarana pembelajaran dalam memahami konsep dasar string menggunakan bahasa Java.

Melalui praktikum ini, penulis mempelajari penggunaan berbagai elemen GUI seperti tombol, label, text field, serta cara mengatur layout agar antarmuka mudah digunakan. Selain itu, penulis juga mempraktikkan bagaimana class dan method mendukung pengolahan input pada GUI. Penulis menyadari bahwa laporan ini masih jauh dari sempurna, baik dari segi penjelasan maupun penyajian. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi perbaikan di masa mendatang. Akhir kata, penulis mengucapkan terima kasih kepada dosen pengampu, asisten praktikum, serta semua pihak yang telah memberikan bimbingan dan dukungan sehingga laporan ini dapat terselesaikan. Semoga laporan ini dapat memberikan manfaat bagi penulis sendiri maupun pembaca.

Padang, 2025

Penulis

## DAFTAR ISI

<b>KATA PENGANTAR.....</b>	ii
<b>DAFTAR ISI .....</b>	iii
<b>BAB 1 PENDAHULUAN.....</b>	4
<b>1.1    Latar Belakang .....</b>	4
<b>1.2    Tujuan .....</b>	4
<b>1.3    Manfaat.....</b>	4
<b>BAB II PEMBAHASAN.....</b>	5
<b>2.1    Operator Aritmatika dengan GUI .....</b>	5
<b>BAB III KESIMPULAN.....</b>	16
<b>3.1    Kesimpulan.....</b>	16
<b>3.2    Saran .....</b>	16
<b>DAFTAR PUSTAKA .....</b>	17

# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

GUI atau *Graphical User Interface* adalah antarmuka berbasis grafis yang memudahkan pengguna berinteraksi dengan program menggunakan ikon dan elemen visual. Jendela merupakan komponen utama sebuah aplikasi GUI. Di dalam jendela kita masukkan berbagai macam komponen seperti tombol, teks, combobox, dan lain-lain. Jendela pada pemrograman Swing dapat kita buat dengan Class JFrame.

JFrame adalah sebuah class yang otomatis ditambahkan ke komponen swing (GUI) milik Java. Ini cocok digunakan ketika ingin membuat aplikasi desktop lengkap dengan menu, tombol, dan komponen lainnya dengan mudah.

### 1.2 Tujuan

1. Memahami cara kerja operator aritmatika di Java dan melihat efek masing-masing operator pada berbagai nilai input.
2. Melatih kemampuan membaca alur eksekusi program, termasuk bagaimana variabel diisi, digunakan, dan ditampilkan kembali.

### 1.3 Manfaat

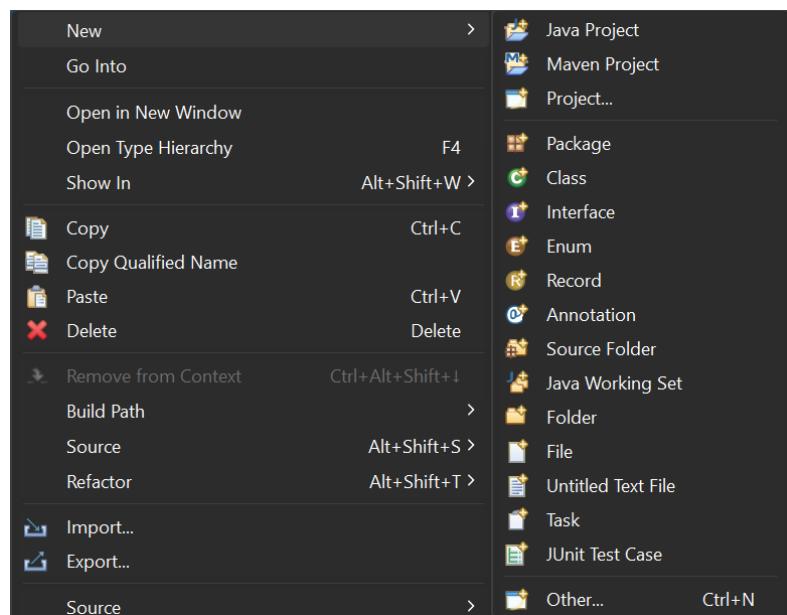
1. Meningkatkan ketelitian logika karena operasi aritmatika adalah fondasi untuk topik lebih kompleks seperti algoritma, perulangan, dan struktur data.
2. Membiasakan diri menggunakan input dari pengguna, sehingga program lebih dinamis dan tidak bergantung pada nilai statis.

## BAB II

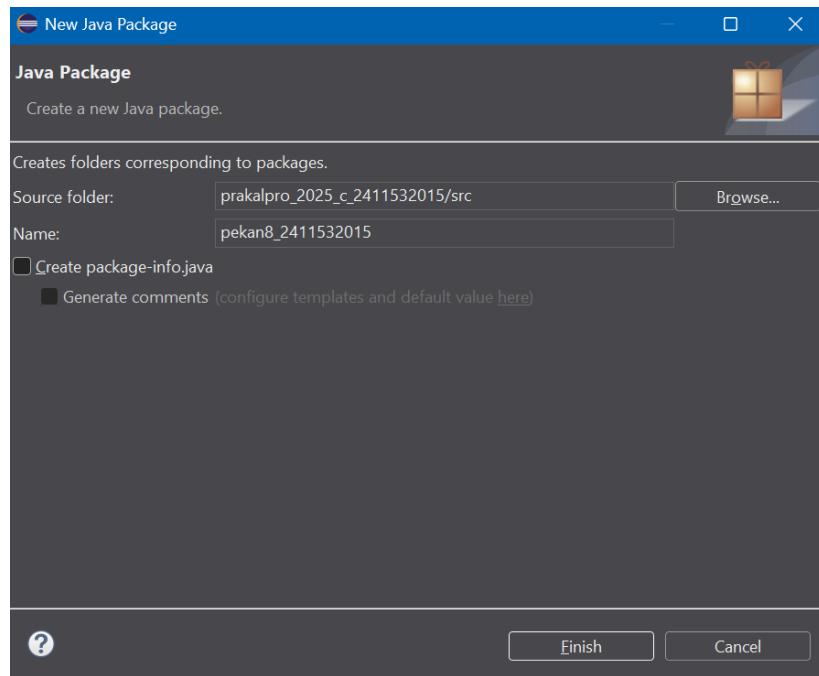
### PEMBAHASAN

#### 2.1 Operator Aritmatika dengan GUI

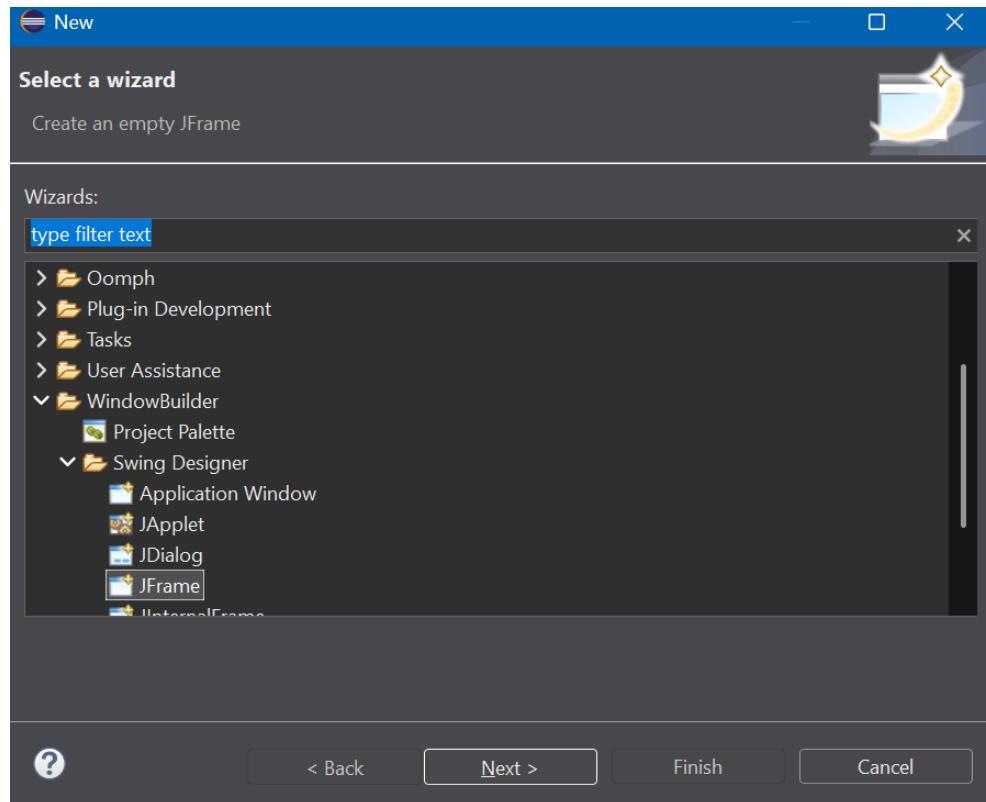
1. Buat Buat *package* baru dengan klik kanan pada src, pilih new lalu *package*



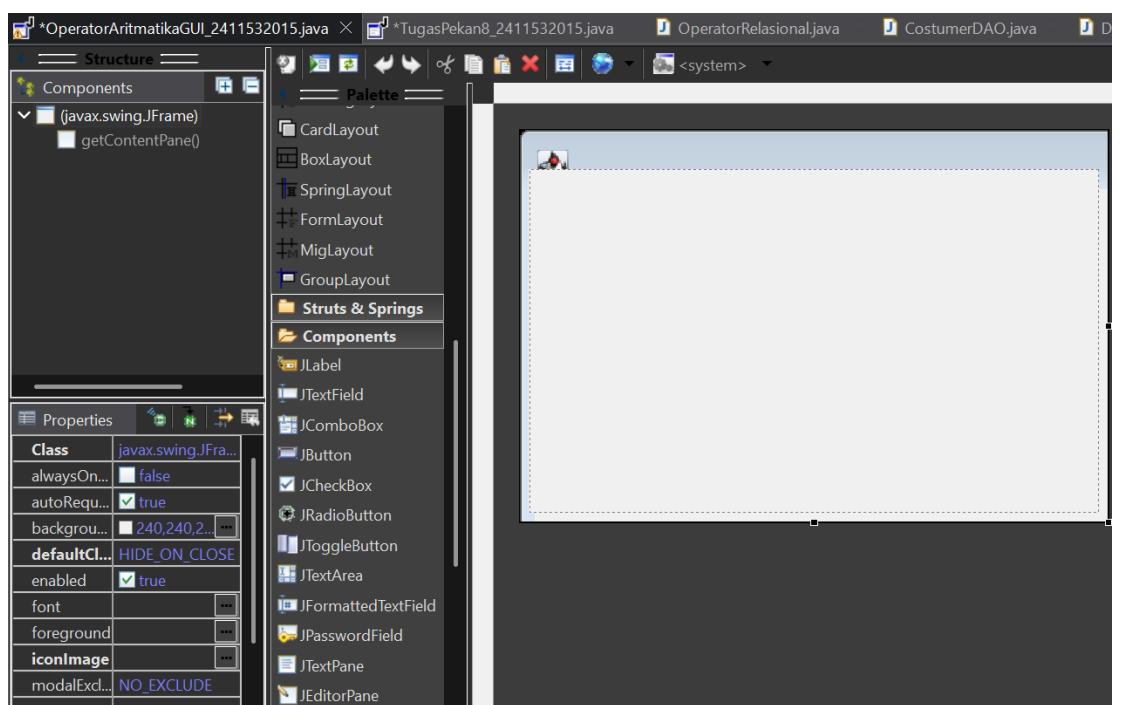
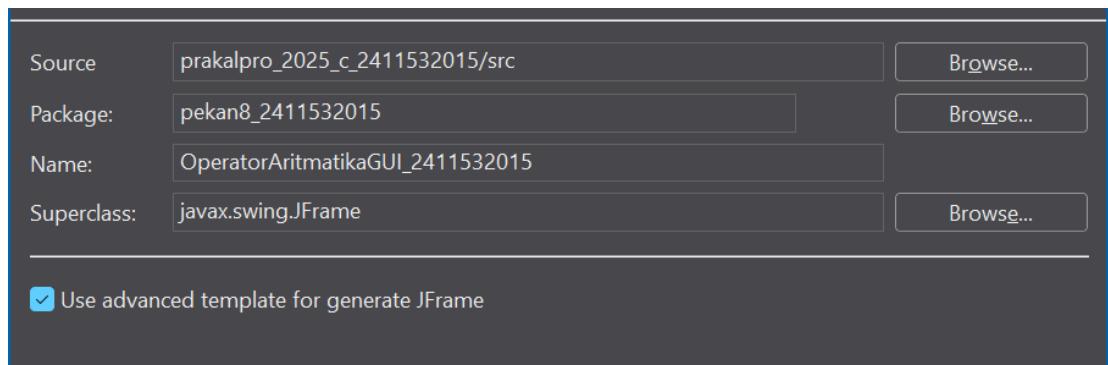
2. Buat *package* dengan nama pekan 8



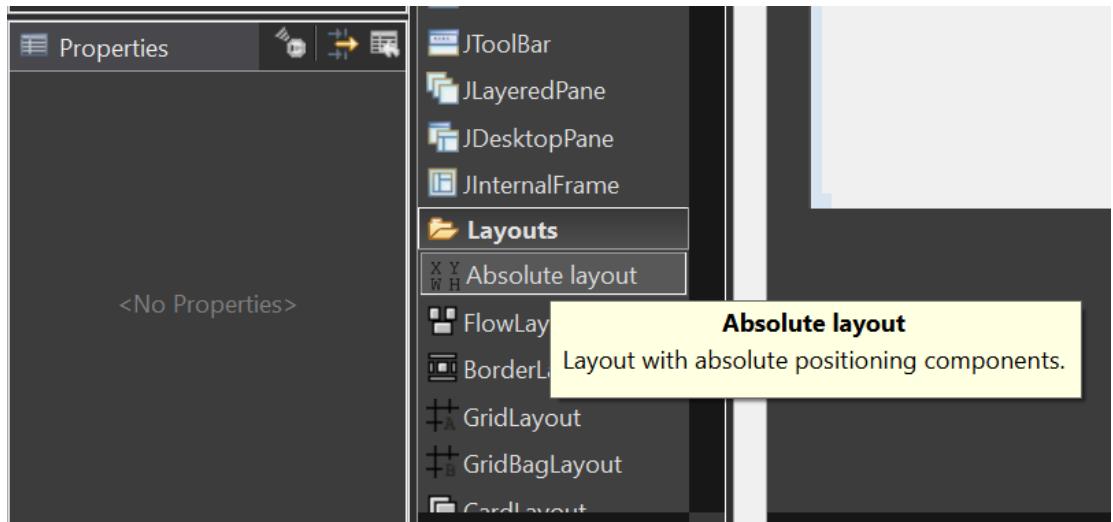
3. Buat *Jframe* dengan cara klik kanan pada package pekan 8 lalu pilih new lalu other dan pilih JFrame



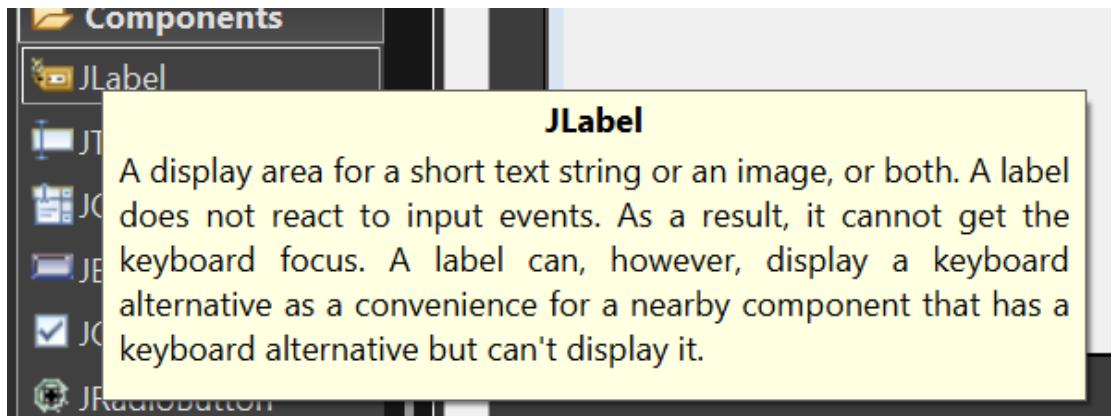
4. Buat *JFrame* dengan nama Operator Aritmatika lalu pergi ke bagian design



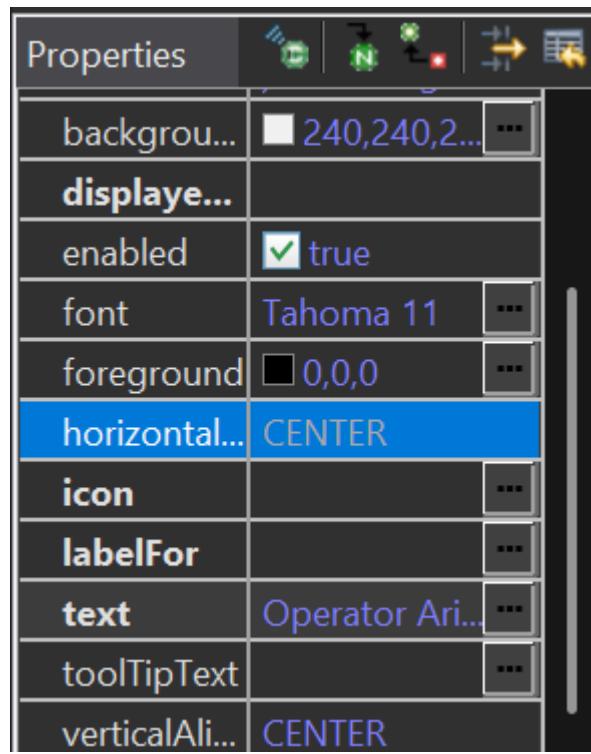
5. Tambahkan absolute layout pada design



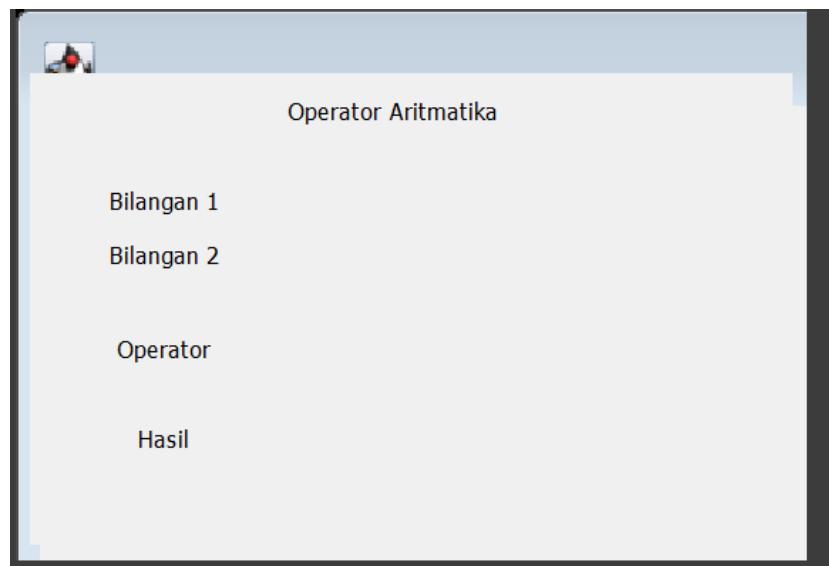
6. Lalu tambahkan JLabel pada design untuk judul



7. Ubah pada bagian text menjadi judul yang diinginkan “Operator Aritmatika”, dan ubah horizontal menjadi center



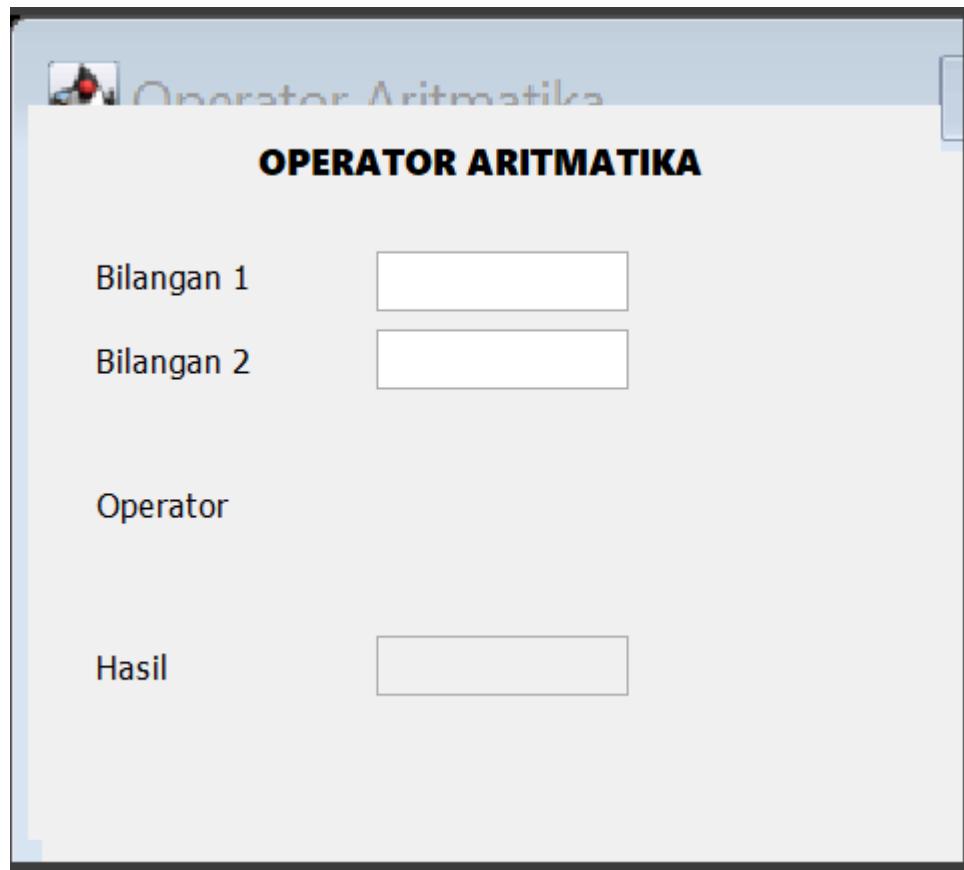
8. Tambahkan lagi JLabel 4 buah dengan nama, Bilangan 1, Bilangan 2, Operator, dan Hasil. Hingga tampilannya nanti seperti ini



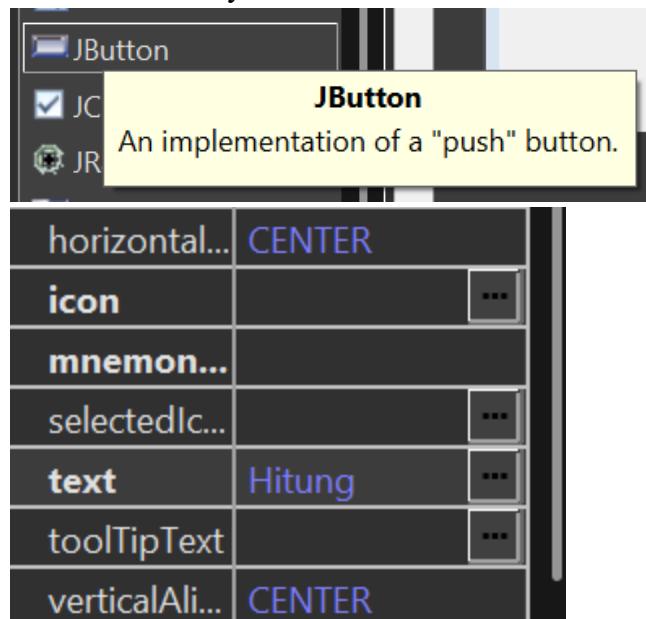
9. Setelah itu tambahkan JTextField ke design untuk memasukkan input bilangan 1 dan 2 serta tempat output hasil



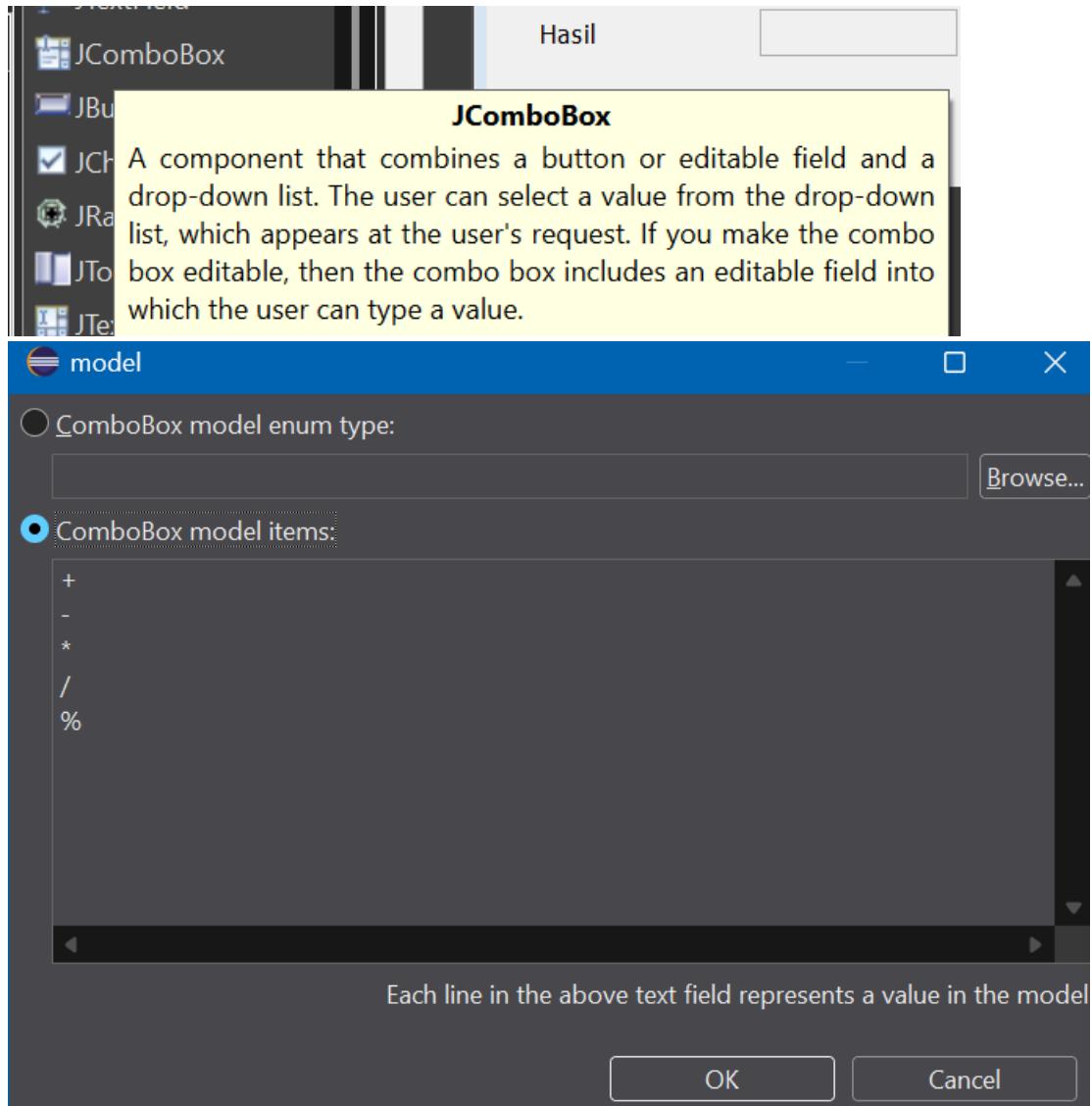
10. Hingga tampilannya seperti ini



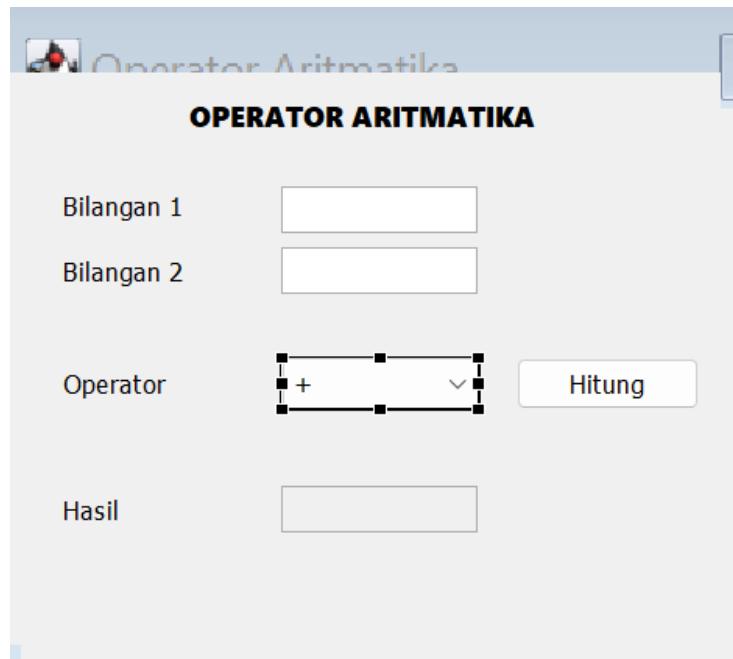
11. Tambahkan juga JButton untuk dengan nama Hitung untuk melakukan operasi aritmatika nantinya



12. Lalu tambahkan JComboBox untuk memilih operator. Masukkan operator pada model



13. Berikut tampilan akhir designnya



14. Setelah itu klik dua kali pada tombol hitung dan masukkan kode berikut

```

int hasil;
public void actionPerformed(ActionEvent e) {
    if (txtBil1.getText().trim().isEmpty()) {
        pesanPeringatan("Bilangan 1 Harus Diisi");
    } else if (txtBil2.getText().trim().isEmpty()) {
        pesanPeringatan("Bilangan 2 harus diisi");
    } else {
        try {
            int a = Integer.valueOf(txtBil1.getText());
            int b = Integer.valueOf(txtBil2.getText());
            int c = cbOperator.getSelectedIndex();

            if (c==0) {
                hasil = a + b;
            }
            if (c==1) {
                hasil = a - b;
            }
            if (c==2) {
                hasil = a * b;
            }
            if (c==3) {
                hasil = a / b;
            }
            if (c==4) {
                hasil = a % b;
            }
            txtHasil.setText(String.valueOf(hasil));
        } catch (NumberFormatException ex) {
            pesanError("Bilangan 1 dan Bilangan 2 harus angka");
        }
    }
}

```

15. Variabel ini untuk menampung hasil perhitungan

```
int hasil;
```

16. Untuk memastikan user memasukkan input tidak kosong

```
if (txtBil1.getText().trim().isEmpty()) {  
    pesanPeringatan("Bilangan 1 Harus Diisi");  
} else if(txtBil2.getText().trim().isEmpty()) {  
    pesanPeringatan("Bilangan 2 harus diisi");
```

17. Blok try catch untuk mengubah isi textfield menjadi angka int

```
}else {  
    try {  
        int a = Integer.valueOf(txtBil1.getText());  
        int b = Integer.valueOf(txtBil2.getText());  
    } catch (Exception e) {  
        pesanError("Bilangan harus angka");  
    }
```

18. Mengambil operator yang dipilih berdasarkan logika aritmatika

```
int c = cbOperator.getSelectedIndex();  
  
if (c==0) {  
    hasil = a + b;  
}  
if (c==1) {  
    hasil = a - b;  
}  
if (c==2) {  
    hasil = a * b;  
}  
if (c==3) {  
    hasil = a / b;  
}  
if (c==4) {  
    hasil = a % b;  
},
```

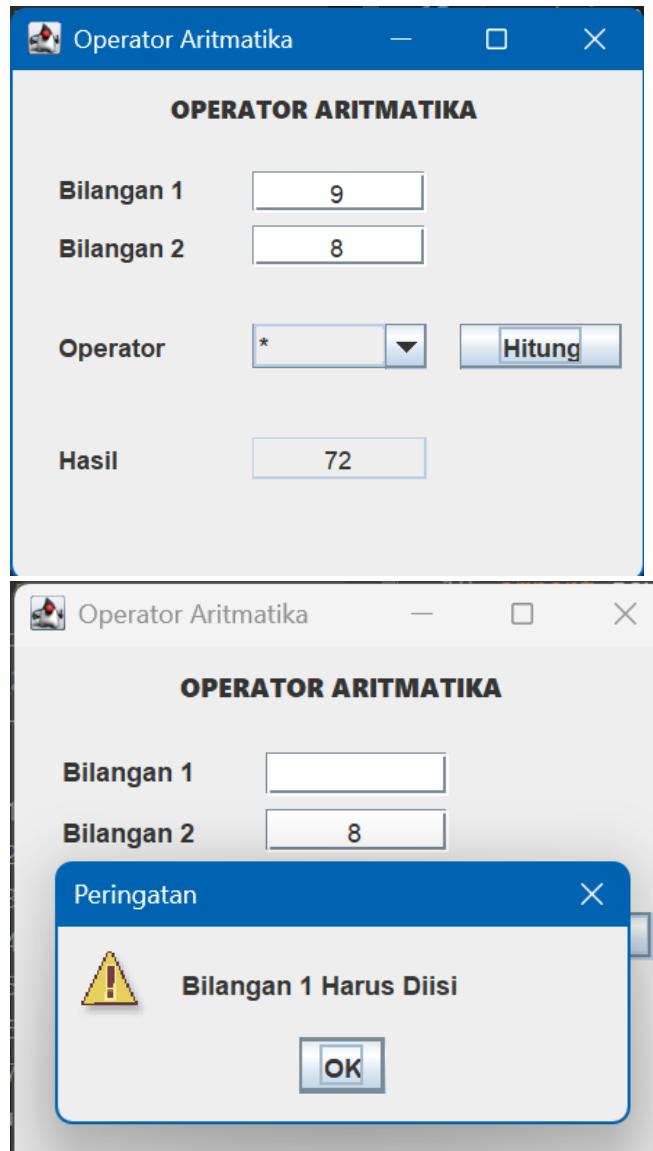
19. Untuk menampilkan hasil serta catch untuk menangkap kesalahan input

```
}  
txtHasil.setText(String.valueOf(hasil));  
  
} catch (NumberFormatException ex) {  
    pesanError("Bilangan 1 dan Bilangan 2 harus angka");  
}
```

20. Tambahkan method berikut pada bagian atas untuk memunculkan dialog warning dan error

```
private void pesanPeringatan(String pesan) {  
    JOptionPane.showMessageDialog(this, pesan, "Peringatan", JOptionPane.WARNING_MESSAGE);  
}  
  
private void pesanError(String pesan) {  
    JOptionPane.showMessageDialog(this, pesan, "Kesalahan", JOptionPane.ERROR_MESSAGE);  
}
```

21. Berikut output ketika dijalankan



## **BAB III**

### **KESIMPULAN**

#### **3.1 Kesimpulan**

Praktikum ini menunjukkan bagaimana bahasa Java memproses operasi matematika dasar seperti penjumlahan, pengurangan, perkalian, pembagian, dan sisa bagi. Dari percobaan, terlihat bahwa Java selalu mengikuti aturan prioritas operator (“operator precedence”), sehingga hasil perhitungan tidak hanya bergantung pada operator yang digunakan, tetapi juga urutannya.

#### **3.2 Saran**

Praktikum ini akan lebih efektif jika peserta tidak hanya mencoba operasi dasar, tetapi juga mengeksplorasi berbagai skenario yang lebih beragam agar memahami perilaku operator secara menyeluruh. Penggunaan komentar yang lebih jelas dan terfokus pada alasan di balik setiap hasil perhitungan juga dapat membantu memperkuat pemahaman konseptual, bukan sekadar menghafal fungsi operator.

## **DAFTAR PUSTAKA**

[1] blogspot, “Class dengan JFrame (Java GUI)”. [Daring]. Tersedia pada:  
<https://javaprogkomp.blogspot.com/2013/06/class-dengan-jframe-java-gui.html>