

Bab. 01

Graphical User Interface (GUI) & Komponennya

A. Konsep Pemrograman GUI

Pemrograman GUI adalah pemrograman yang mengandalkan kemampuan tampilan dalam bentuk grafik, dimana program yang dibuat dapat memberikan kenikmatan tersendiri pada penggunanya, tidak lagi monoton dengan hitam putih dan mode dos.

Dalam pemrograman GUI, kita dapat menggunakan komponen GUI untuk menciptakan sebuah tampilan atau interface yang diinginkan dengan menyertakan package yang kita butuhkan didalam program yang kita buat. Package tersebut diantaranya AWT (Application Windowing Toolkit) dan Swing.

AWT adalah GUI toolkit pertama pada Bahasa pemrograman java, namun package ini kurang komponen yang bisa digunakan untuk membangun sebuah aplikasi desktop. AWT merupakan aplikasi native, dimanan jika aplikasi yang dibangun dengan package AWT dijalankan pada sistem operasi windows maka aplikasi tersebut akan terlihat seperti aplikasi windows pada umumnya, begitu pula jika dijalankan pada sistem operasi Mac maupun GNU/Linux. Hal ini karena AWT benar-benar memanggil native subrutin untuk menggambar setiap komponennya ke layar.

Swing adalah package/library java yang digunakan untuk menciptakan grafik User Interface (GUI), dengan java swing kita dapat membuat user interface yang cross platform atau OS independent. Cross platform artinya user interface yang kita buat dapat dijalankan pada sistem operasi apa saja (OS yang support Java) dengan tampilan yang relative sama. Kita dapat membuat user interface yang menyerupai windows XP, MAC atau Linux tanpa tergantung dengan sistem operasi yang kita gunakan. Swing adalah bagian dari Java Foundation Classes (JFC) yang mana terdapat fasilitas untuk menambahkan Rich Graphical Functionality.

Tipe program yang berbasis pada GUI komponen, diantaranya :

1. Graphical componens (GUI)
Yaitu berupa rancangan tampilan program yang dikemas dengan menggunakan komponen-komponen grafik
2. Listener Methods
Yaitu berfungsi untuk menerima dan merespon event yang terjadi
3. Application Methods
Fungsi atau baris perintah yang berguna bagi pengguna untuk menghasilkan kegiatan dalam bentuk respon.

B. Class / Komponen GUI

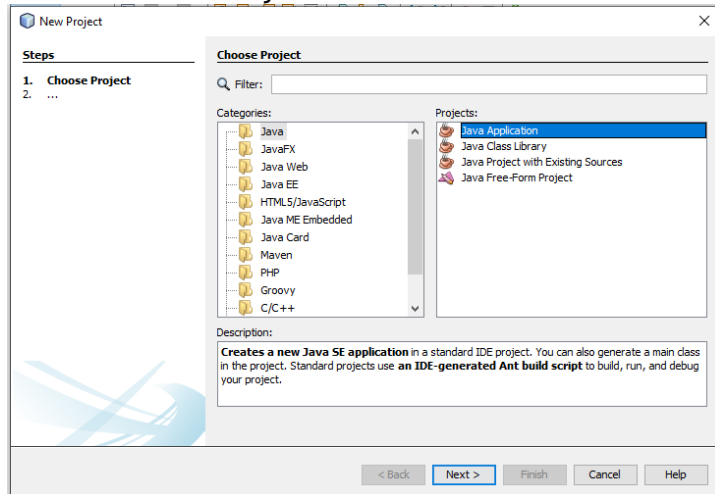
Beberapa komponen GUI diataranya :

[illegible]

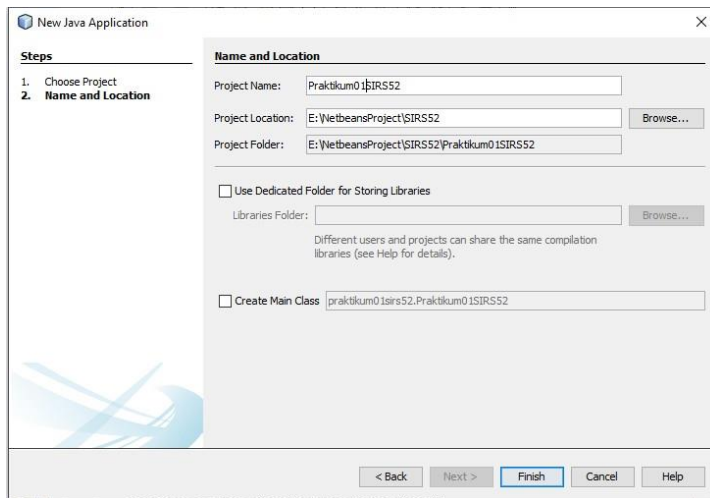
Bab. 02

Text Field & Button & JOptionPane

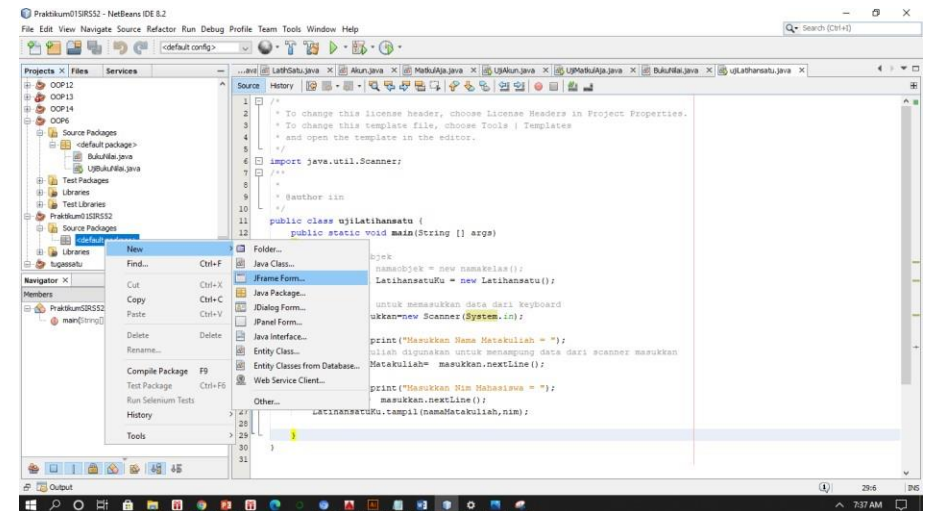
1. Klik File → New Project



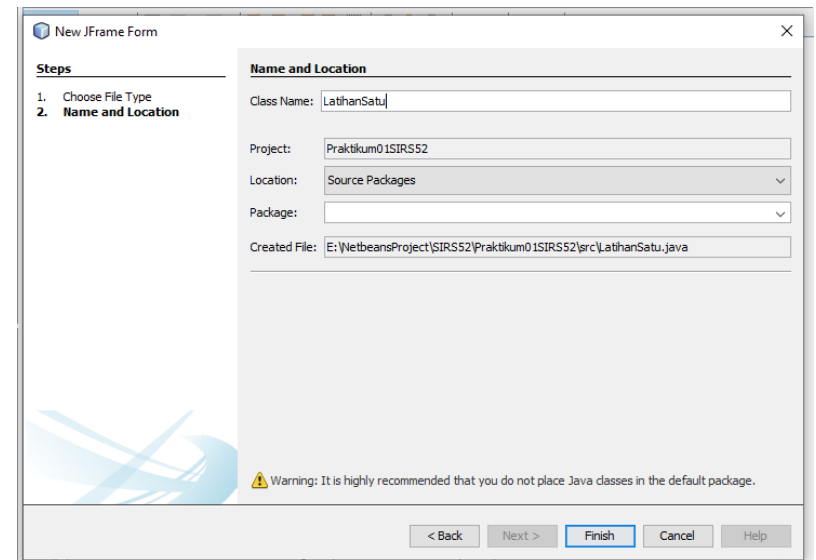
2. Tulis nama project dan lokasi penyimpanan



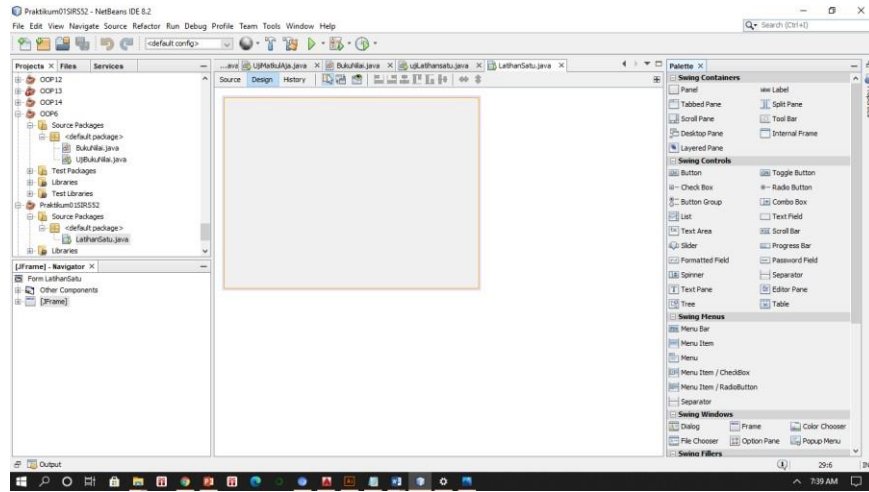
3. Klik kanan default package



4. Buat nama form



5. Berikut jendela tampilan form



6. Atur komponen yang akan digunakan

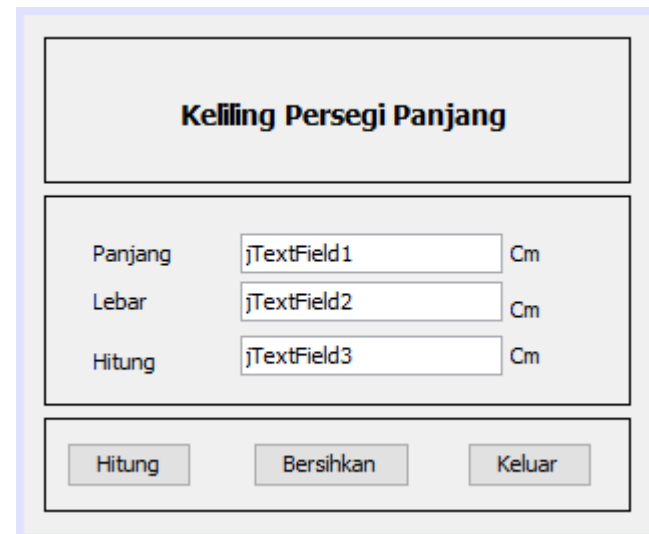


7. Ubah Propertise dari komponen

| No. | Komponen | Propertise & Code |
|-----|----------|---|
| 1. | jPanel1 | Border = Line Border |
| 2. | jPanel2 | Border = Line Border |
| 3. | jPanel3 | Border = Line Border |
| 4. | jLabel1 | Font (CTRL+Space) = Tahoma, Bold, 14 Text = Keliling Persegi Panjang |

| | | |
|----|-------------|---|
| 5 | jLabel2 | Text = Panjang |
| 6 | jLabel3 | Text = Lebar |
| 7 | jLabel4 | Text = Hitung |
| 8 | jLabel5 | Text = Cm |
| 9 | jLabel6 | Text = Cm |
| 10 | jLabel7 | Text = Cm |
| 11 | jButton1 | Propertise :Text = Hitung Code :Variabel Name = Hitung |
| 12 | jButton2 | Propertise : Text = Bersihkan Code : Variabel Name =Bersihkan |
| 13 | jButton3 | Propertise : Text = Keluar Code : Variabel Name = Keluar |
| 14 | JFrame | Propertise : Title = Program Hitung Keliling Code : Form Size = [329, 329] Generate Size = true Generate Center =true |
| 15 | jTextField1 | Code : Variabel Nama= panjang |
| 16 | jTextField2 | Code : Variabel Nama= lebar |
| 17 | jTextField3 | Code : Variabel Nama= hasil |

8. Hasil perubahan properties



9. Kode program untuk JButton3 “Keluar” adalah :

```
private void KeluarActionPerformed(java.awt.event.ActionEvent  
evt) {  
    // TODO add your handling code here:  
    int jawab = JOptionPane.showOptionDialog(this, "Ingin  
Keluar?", "Keluar",  
JOptionPane.YES_NO_OPTION, JOptionPane.QUESTION_MESSAGE, nul  
l, null, null)  
    if (jawab == JOptionPane.YES_OPTION)  
    {  
        JOptionPane.showMessageDialog(this, "Program Akan  
Keluar");  
        System.exit(0);  
    }  
}
```

10. Buat sebuah method “Kosongkan_nilai” ?

```
private void kosongkan_nilai()  
{  
    panjang.setText("");  
    lebar.setText("");  
    hasil.setText("");  
    panjang.requestFocus();  
}
```

11. Klik 2x pada Button Bersihkan dan panggil method “Kosongkan_nilai”

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

12. Untuk menghapus tampilan JTextbox panggil method “Kosongkan Nilai” pada method LatihanSatu()!

```
public LatihanSatu()  
{  
    initComponents();  
    kosongkan_nilai();  
}
```

13. Buat method “Hitung” !


```
private void hitung()  
{  
    float panjangpp, lebarpp, hasilpp;  
    panjangpp=Float.valueOf(panjang.getText());  
    lebarpp=Float.valueOf(lebar.getText());  
    hasilpp=2*(panjangpp+lebarpp);  
    hasil.setText(String.valueOf(hasilpp));  
}
```

14. Panggil method hitung pada Button “Hitung” !

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

Tugas :

1. Buatlah program dibawah ini !

 - □ ×

Kalkulator

X27

Keluar

Y

Hapus

Hasil3

=

/

-

+

*

^

√

$\sqrt[3]{}$

x^y

x^2

Bab. 03

Array List memanfaatkan JRadioButton, JCheckBox dan JTable

A. ArrayList

ArrayList pada bahasa pemrograman Java, merupakan salah satu collection yang bisa kalian gunakan untuk menampilkan daftar atau list nilai/value, yang bersifat dinamis dan juga dapat dimodifikasi, ukuran pada ArrayList dapat kita ubah dengan cara menambahkan (add) atau menghapusnya (remove). Tidak seperti Array pada umumnya, pada ArrayList kita tidak dapat menggunakan tipe data primitive, seperti int, char, boolean, float, dsb.

B. JRadioButton

Radio Button adalah suatu tombol yang digunakan untuk mengambil salah satu pilihan dari banyak pilihan yang tersedia

Radio Button Group berfungsi untuk merelasikan antara beberapa radio button agar bisa di seleksi salah satu saja.

C. JCheckBox

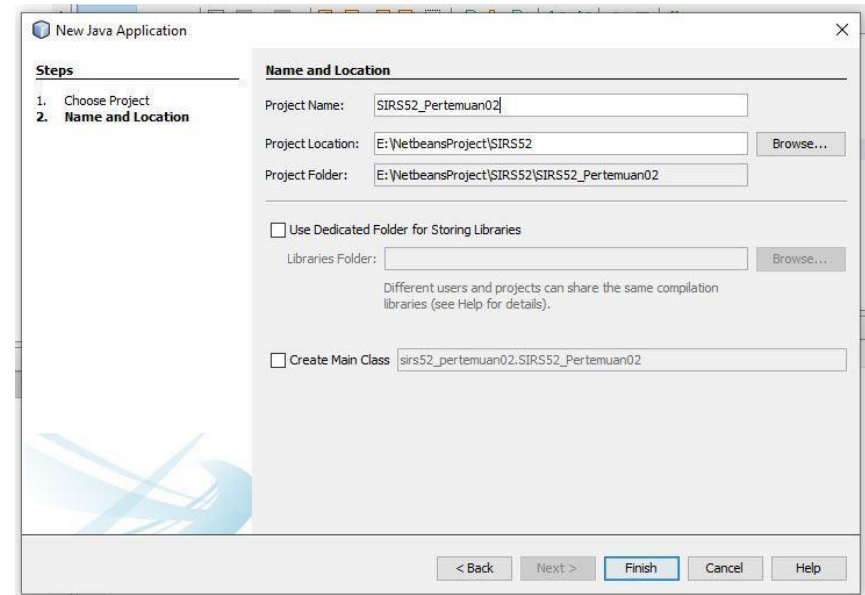
JCheckBox adalah komponen yang digunakan untuk memilih pilihan lebih dari satu (*multiple selection*)

D. JTable

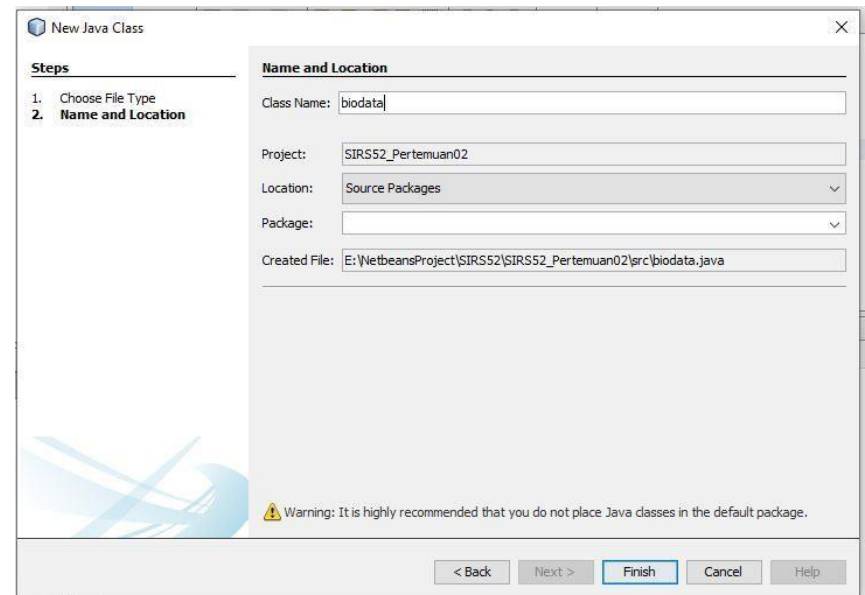
JTable adalah komponen yang digunakan untuk menampilkan data pada table dalam jumlah banyak

E. Praktikum

1. Buatlah project seperti dibawah ini !



2. Buatlah java class berikut !



3. Isi code berikut pada java class biodata

```
public class biodata {
    private String namaMhs, alamatMhs, jkelaminMhs, nimMhs;
    public biodata(){}
    public biodata(String nim,String nama, String alamat, String
jkelamin)
    {
        this.namaMhs= nama;
        this.alamatMhs=alamat;
        this.jkelaminMhs=jkelamin;
        this.nimMhs=nim;
    }
    public String getnama()
    {
        return namaMhs;
    }
    public String getalamat()
    {
        return alamatMhs;
    }
    public String getnim()
    {
        return nimMhs;
    }
    public String getjkelamin()
    {
        return nimMhs;
    }
}
```

4. Buatlah JFrameForm “InputBiodata” dengan tampilan form sebagai berikut !

5. Atur properties sebagai berikut :

| No. | Komponen | Propertise | Isi / Pilihan |
|-----|---------------|-------------|--|
| 1 | JLabel1 | Text | Hobi |
| 2 | JLabel2 | Text | STMIK Profesional Makassar |
| 3 | JLabel3 | Text | NIM |
| 4 | JLabel4 | Text | Nama |
| 5 | JLabel5 | Text | Alamat |
| 6 | JLabel6 | Text | Jenis Kelamin |
| 7 | JLabel7 | Text | www.stmikprofesional.ac.id |
| 8 | JLabel8 | Text | Jl. AP. Pettarani No. 27 Makassar |
| 9 | ButtonGroup1 | | |
| 10 | JRadioButton1 | Text | Laki-Laki |
| | | ButtonGroup | ButtonGroup1 |
| 11 | JRadioButton1 | Text | Perempuan |
| | | ButtonGroup | ButtonGroup1 |
| 12 | JCheckBox1 | Text | Menggambar |
| 13 | JCheckBox2 | Text | Membaca |
| 14 | JCheckBox3 | Text | Coding |

| | | | |
|----|----------|-------|---------------|
| 15 | JButton1 | Text | SimpanData |
| 16 | JButton2 | Text | Bersihkan |
| 17 | JButton3 | Text | Keluar |
| 18 | JButton5 | Text | HapusData |
| 19 | JTable1 | | |
| 20 | JFrame1 | Title | Input Biodata |

6. Tampilan Antarmuka !

7. Buat kode program untuk metode kosongkan

```

public void kosongkan()
{
    jTextField1.setText("");
    jTextField2.setText("");
    jTextField3.setText("");
    jRadioButton1.setSelected(false);
    jRadioButton2.setSelected(false);
    jCheckBox1.setSelected(false);
    jCheckBox2.setSelected(false);
    jCheckBox3.setSelected(false);
}

```

8. Atur penggunaan metode kosongkan pada class InputBiodata

```

public InputBiodata() {
    initComponents();
    kosongkan();
}

```

9. Atur penggunaan metode kosongkan pada JButton2

```

private void
jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    kosongkan();
}

```

10. Kode program JButton3

```

private void
jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int jawab = JOptionPane.showOptionDialog(this, "Ingin
Keluar?", "Keluar",
JOptionPane.YES_NO_OPTION, JOptionPane.QUESTION_MESSAGE, nul
l, null, null);
    if (jawab == JOptionPane.YES_OPTION)
    {
        JOptionPane.showMessageDialog(this, "Program Akan
Keluar");
        System.exit(0);
    }
}

```

11. Buat array list dan model

```

ArrayList <biodata> listdata=new ArrayList<biodata>();
DefaultTableModel tabelModel;

```


12. Buatlah metode tampilData() !

```
public void tampilData()
{
    String [] kolom={"nim","nama","alamat","jkelamin"};

    Object[][] objData = new Object[listdata.size()][4]; // 3
    merupakan jumlah kolom

    int i = 0;
    for(biodata n : listdata){
        String[] arrayData = {n.getnim(),n.getnama(),
        n.getalamat(), n.getjkelamin()};
        objData[i] = arrayData;
        i++;
    }
    tabelModel =new DefaultTableModel(objData,kolom){
        //membuat tabel tidak bisa diedit saat diklik
        @Override
        public boolean isCellEditable(int rowIndex, int colIndex) {
            return false;
        }
    };
    jTable1.setModel(tabelModel);
}
```

12. Buat metode isidata

```
public void isidata(String nim,String nama,String alamat,String
jkelamin)
{
    biodata data=new biodata(nim,nama,alamat,jkelamin);
    listdata.add(data);
}
```

13. Buat metode pengambilandata

```
private void pengambilandata()
{
    String nim, nama,alamat,jkelamin;
    nim=jTextField1.getText();
    nama=jTextField2.getText();
    alamat=jTextField3.getText();
    //jkelamin
    if (jRadioButton1.isSelected())
        {jkelamin="Laki-laki";}
    else
        {jkelamin="Perempuan";}

    JOptionPane.showMessageDialog(this,"Nim : "+nim+"\n Nama :
    "+nama+"\n Alamat : "+alamat+"\n"+"Jenis Kelamin :
    "+jkelamin,"Info",JOptionPane.INFORMATION_MESSAGE);

    isidata(nim,nama,alamat,jkelamin);
}
```

14. Buatlah

```
private void hapusData()
{
    int hapus=jTable1.getSelectedRow();
    if(hapus!=-1)
    {}
    else
    {
        listdata.remove(hapus);
    }
}
```

kosongkan();

}

15. Buatlah kode berikut pada jButton1

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

16. Buatlah kode program pada jButton5 berikut

```
private void
jButton5ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    hapusData();
    tampilData();
}
```

17. Buatlah kode program pada jButton1 berikut :

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

18. Tambahkan pemanggilan metode tampilData pada kelas inputbiodata

```
public InputBiodata() {
    initComponents();
    tampilData();
}
```