**LAPORAN PRAKTIKUM**

**BAHASA PEMROGRAMAN 2**

**MODUL 6**

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](#_Toc197545827)

[BAB I 1](#_Toc197545828)

[PRETEST 1](#_Toc197545829)

[Tuliskan Struktur Program Applet Java 1](#_Toc197545830)

[BAB II 3](#_Toc197545831)

[PRAKTIKUM 3](#_Toc197545832)

[Applet1 **Kesalahan! Bookmark tidak ditentukan.**](#_Toc197545833)

[AppletMouse **Kesalahan! Bookmark tidak ditentukan.**](#_Toc197545834)

[Applet3D **Kesalahan! Bookmark tidak ditentukan.**](#_Toc197545835)

[BAB III 21](#_Toc197545836)

[POSTTEST 21](#_Toc197545837)

[Buatlah File Applet Perhitungan Matematika **Kesalahan! Bookmark tidak ditentukan.**](#_Toc197545838)

[BAB IV 23](#_Toc197545839)

[TUGAS 23](#_Toc197545840)

[Buatlah File Applet Animasi dari File .gif 23](#_Toc197545841)

[BAB V 47](#_Toc197545842)

[KESIMPULAN 47](#_Toc197545843)

# BAB I

# PRETEST

## Tuliskan Struktur Program Applet Java

Kerjakan pda selembar kertas dengan waktu 5 menit dan dikumpulkan melalui Asisten lab.

import java.applet.Applet;

import java.awt.Graphics;

// Kelas Applet yang merupakan turunan dari kelas Applet

public class MyApplet extends Applet {

// Metode ini dipanggil pertama kali saat applet dimuat

@Override

public void init() {

// Inisialisasi komponen atau variabel

System.out.println("Applet diinisialisasi");

}

// Metode ini dipanggil setiap kali applet dijalankan

@Override

public void start() {

System.out.println("Applet dimulai");

}

// Metode ini digunakan untuk menggambar grafik atau teks di applet

@Override

public void paint(Graphics g) {

g.drawString("Hello, Applet!", 20, 20);

System.out.println("Applet digambar");

}

// Metode ini dipanggil ketika applet dihentikan sementara

@Override

public void stop() {

System.out.println("Applet dihentikan sementara");

}

// Metode ini dipanggil ketika applet ditutup atau pengguna meninggalkan halaman

@Override

public void destroy() {

System.out.println("Applet dihancurkan");

}

}

# BAB II

# PRAKTIKUM

## Input Data

import javax.swing.table.DefaultTableModel;

import javax.swing.JOptionPane;

import java.sql.Statement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.PreparedStatement; // Ditambahkan untuk PreparedStatement

/\*\*

\*

\* @author Muhammad Rizal Nur F

\*/

public class FUser extends javax.swing.JFrame {

Statement st;

ResultSet rs;

Koneksi koneksi;

// Ganti jTextField2 menjadi INamaUser (Anda mungkin perlu melakukannya di Design View NetBeans)

// Jika tidak, pastikan Anda telah mengganti nama variabel jTextField2 menjadi INamaUser di bagian deklarasi variabel di bawah.

// Untuk contoh ini, saya akan mengasumsikan Anda telah mengubahnya menjadi INamaUser.

public FUser() {

koneksi = new Koneksi();

initComponents();

load\_data();

IDOtomatis(); // Panggil IDOtomatis

level(); // Panggil level (jika Anda ingin mengisi CLevel dari DB)

// Jika tidak, biarkan CLevel diisi manual seperti di initComponents

bersih(); // Bersihkan form dan siapkan ID baru

IIdUser.setEditable(false); // ID User tidak boleh diedit manual

}

private void load\_data() {

Object header[] = {"ID USER", "LEVEL", "NAMA USER", "JENIS KELAMIN",

"NO HANDPHONE", "USERNAME", "PASSWORD"};

DefaultTableModel data = new DefaultTableModel(null, header);

TUser.setModel(data);

String sql = "SELECT tbl\_user.id\_user, tbl\_level.level, tbl\_user.nama\_user, "

+ "tbl\_user.jk, tbl\_user.NOPE, tbl\_user.username, tbl\_user.password "

+ "FROM tbl\_user INNER JOIN tbl\_level ON tbl\_user.id\_level = tbl\_level.id\_level ORDER BY tbl\_user.id\_user ASC;"; // Tambahkan ORDER BY

try {

if (koneksi.con == null) {

JOptionPane.showMessageDialog(null, "Koneksi ke database gagal!");

return;

}

st = koneksi.con.createStatement();

rs = st.executeQuery(sql);

while (rs.next()) {

String k1 = rs.getString(1);

String k2 = rs.getString(2);

String k3 = rs.getString(3);

String k4 = rs.getString(4);

String k5 = rs.getString(5);

String k6 = rs.getString(6);

String k7 = rs.getString(7);

String k[] = {k1, k2, k3, k4, k5, k6, k7};

data.addRow(k);

}

} catch (SQLException e) {

JOptionPane.showMessageDialog(null, "Error saat load data: " + e.getMessage());

e.printStackTrace();

}

}

// Implementasi dari gambar

private void IDOtomatis() {

try {

if (koneksi.con == null) {

JOptionPane.showMessageDialog(this, "Koneksi Database Gagal!");

return;

}

st = koneksi.con.createStatement();

// Gunakan id\_user sesuai dengan yang ada di load\_data

String sql = "SELECT \* FROM tbl\_user order by id\_user desc";

rs = st.executeQuery(sql);

if (rs.next()) {

String idUser = rs.getString("id\_user").substring(4); // Ambil angka setelah "USER"

String AN = "" + (Integer.parseInt(idUser) + 1);

String Nol = "";

if (AN.length() == 1) {

Nol = "00";

} else if (AN.length() == 2) {

Nol = "0";

} else if (AN.length() == 3) {

Nol = "";

}

// Sesuaikan prefix jika berbeda, di gambar adalah "USER"

IIdUser.setText("USER" + Nol + AN);

} else {

IIdUser.setText("USER001");

}

} catch (SQLException e) {

JOptionPane.showMessageDialog(null, "Error ID Otomatis: " + e.getMessage());

e.printStackTrace();

}

}

// Method untuk mengisi ComboBox Level (jika dari database)

// Jika Anda sudah mengisi CLevel secara manual di initComponents, method ini bisa di-skip

// atau disesuaikan. Untuk saat ini, kita anggap CLevel sudah diisi dari Design.

private void level() {

// CLevel.removeAllItems(); // Hapus item lama jika diisi dari DB

// CLevel.addItem("- Pilih Level -"); // Item default

// try {

// if (koneksi.con == null) return;

// st = koneksi.con.createStatement();

// String sql = "SELECT level FROM tbl\_level ORDER BY level ASC";

// rs = st.executeQuery(sql);

// while(rs.next()){

// CLevel.addItem(rs.getString("level"));

// }

// } catch (SQLException e) {

// JOptionPane.showMessageDialog(null, "Error memuat level: "+e.getMessage());

// }

// Karena CLevel sudah diisi di initComponents, kita bisa biarkan kosong atau

// hanya memastikan ada item yang terpilih jika perlu.

if (CLevel.getItemCount() > 0) {

CLevel.setSelectedIndex(0); // Pilih item pertama (misal "ADMIN")

}

}

private void bersih() {

// IIdUser.setText(""); // ID User akan di-set oleh IDOtomatis()

if (CLevel.getItemCount() > 0) {

CLevel.setSelectedIndex(0);

}

INamaUser.setText(""); // Pastikan jTextField2 sudah di-rename menjadi INamaUser

Gjk.clearSelection(); // Membersihkan pilihan radio button

INope.setText("");

IUser.setText("");

IPassword.setText("");

INamaUser.requestFocus(); // Fokus ke Nama User

IDOtomatis(); // Panggil lagi untuk ID baru setelah simpan/bersih

}

private int getIdLevelByName(String levelName) {

// Fungsi ini untuk mendapatkan id\_level berdasarkan nama level

// Ini asumsi, sesuaikan dengan struktur tabel tbl\_level Anda

// Misalnya: ADMIN -> 1, GUDANG -> 2, KASIR -> 3

// Cara yang lebih baik adalah query ke tbl\_level

try {

if (koneksi.con == null) return -1; // Indikasi error

String sql = "SELECT id\_level FROM tbl\_level WHERE level = ?";

PreparedStatement ps = koneksi.con.prepareStatement(sql);

ps.setString(1, levelName);

ResultSet rsLevel = ps.executeQuery();

if (rsLevel.next()) {

return rsLevel.getInt("id\_level");

}

} catch (SQLException e) {

JOptionPane.showMessageDialog(this, "Error mendapatkan ID Level: " + e.getMessage());

e.printStackTrace();

}

// Fallback jika tidak ketemu atau error (sebaiknya tidak terjadi jika data konsisten)

// Atau bisa juga return -1 dan handle errornya

if (levelName.equalsIgnoreCase("ADMIN")) return 1;

if (levelName.equalsIgnoreCase("GUDANG")) return 2;

if (levelName.equalsIgnoreCase("KASIR")) return 3;

return 0; // Default atau error

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents

private void initComponents() {

Gjk = new javax.swing.ButtonGroup();

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

jLabel3 = new javax.swing.JLabel();

jLabel4 = new javax.swing.JLabel();

jLabel5 = new javax.swing.JLabel();

jLabel6 = new javax.swing.JLabel();

jLabel7 = new javax.swing.JLabel();

jLabel8 = new javax.swing.JLabel();

IIdUser = new javax.swing.JTextField();

INamaUser = new javax.swing.JTextField(); // Diganti dari jTextField2

INope = new javax.swing.JTextField();

IUser = new javax.swing.JTextField();

CLevel = new javax.swing.JComboBox<String>();

JkL = new javax.swing.JRadioButton();

JkP = new javax.swing.JRadioButton();

IPassword = new javax.swing.JPasswordField();

jLabel9 = new javax.swing.JLabel();

jScrollPane1 = new javax.swing.JScrollPane();

TUser = new javax.swing.JTable();

jLabel10 = new javax.swing.JLabel();

BSimpan = new javax.swing.JButton(); // Diganti dari jButton1

BEdit = new javax.swing.JButton(); // Diganti dari jButton2

BHapus = new javax.swing.JButton(); // Diganti dari jButton3

Gjk.add(JkL);

Gjk.add(JkP);

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

setTitle("Form User"); // Menambahkan Judul Window

jLabel1.setFont(new java.awt.Font("Segoe UI", 3, 12)); // NOI18N

jLabel1.setText("FORM INPUT DATA");

jLabel2.setText("ID USER");

jLabel3.setText("LEVEL");

jLabel4.setText("NAMA USER");

jLabel5.setText("JENIS KELAMIN");

jLabel6.setText("NO HP");

jLabel7.setText("USERNAME");

jLabel8.setText("PASSWORD");

CLevel.setModel(new javax.swing.DefaultComboBoxModel(new String[] { "ADMIN", "GUDANG", "KASIR" }));

JkL.setText("L");

JkL.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

JkLActionPerformed(evt);

}

});

JkP.setText("P");

jLabel9.setFont(new java.awt.Font("Segoe UI", 3, 18)); // NOI18N

jLabel9.setText("DATA USER APLIKASI KASIR");

TUser.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

{null, null, null, null},

{null, null, null, null},

{null, null, null, null},

{null, null, null, null}

},

new String [] {

"Title 1", "Title 2", "Title 3", "Title 4"

}

));

jScrollPane1.setViewportView(TUser);

jLabel10.setFont(new java.awt.Font("Segoe UI", 3, 12)); // NOI18N

jLabel10.setText("DATA USER");

BSimpan.setText("SIMPAN");

BSimpan.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

BSimpanActionPerformed(evt);

}

});

BEdit.setText("EDIT");

BEdit.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

BEditActionPerformed(evt);

}

});

BHapus.setText("HAPUS");

BHapus.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

BHapusActionPerformed(evt);

}

});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(42, 42, 42)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(135, 135, 135)

.addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel10)

.addGap(236, 236, 236))

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jLabel3)

.addComponent(jLabel4)

.addComponent(jLabel5)

.addComponent(jLabel6)

.addComponent(jLabel7)

.addComponent(jLabel8)

.addComponent(jLabel2))

.addGap(25, 25, 25)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(BSimpan)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(BEdit)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(BHapus)

.addGap(0, 0, Short.MAX\_VALUE))

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addGroup(layout.createSequentialGroup()

.addComponent(JkL)

.addGap(18, 18, 18)

.addComponent(JkP))

.addComponent(CLevel, javax.swing.GroupLayout.PREFERRED\_SIZE, 123, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(IIdUser, javax.swing.GroupLayout.DEFAULT\_SIZE, 206, Short.MAX\_VALUE)

.addComponent(INope)

.addComponent(IUser)

.addComponent(IPassword)

.addComponent(INamaUser)) // Diganti dari jTextField2

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 127, Short.MAX\_VALUE)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 630, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 34, Short.MAX\_VALUE))))))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jLabel9)

.addGap(350, 350, 350)) // Disesuaikan agar lebih ke tengah

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGap(20, 20, 20)

.addComponent(jLabel9)

.addGap(32, 32, 32)

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

.addComponent(jLabel1)

.addComponent(jLabel10))

.addGap(9, 9, 9)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addGroup(layout.createSequentialGroup()

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

.addComponent(jLabel2)

.addComponent(IIdUser, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

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

.addComponent(jLabel3)

.addComponent(CLevel, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

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

.addComponent(jLabel4)

.addComponent(INamaUser, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)) // Diganti dari jTextField2

.addGap(18, 18, 18)

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

.addComponent(jLabel5)

.addComponent(JkL)

.addComponent(JkP))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addComponent(jLabel6)

.addGap(6, 6, 6))

.addComponent(INope, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

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

.addComponent(jLabel7)

.addComponent(IUser, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 141, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(18, 18, 18)

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

.addComponent(jLabel8)

.addComponent(IPassword, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(32, 32, 32)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(BSimpan)

.addComponent(BEdit)

.addComponent(BHapus))

.addContainerGap(117, Short.MAX\_VALUE))

);

pack();

setLocationRelativeTo(null); // Menengahkan window saat muncul

}// </editor-fold>//GEN-END:initComponents

private void JkLActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_JkLActionPerformed

// TODO add your handling code here:

}//GEN-LAST:event\_JkLActionPerformed

private void BSimpanActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_BSimpanActionPerformed

if (koneksi.con == null) {

JOptionPane.showMessageDialog(this, "Koneksi Database Gagal!");

return;

}

String idUser = IIdUser.getText();

String levelName = CLevel.getSelectedItem().toString();

String namaUser = INamaUser.getText(); // Pastikan ini INamaUser

String jk = "";

if (JkL.isSelected()) {

jk = "L";

} else if (JkP.isSelected()) {

jk = "P";

}

String nope = INope.getText();

String username = IUser.getText();

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

// Validasi input dasar

if (namaUser.isEmpty() || jk.isEmpty() || nope.isEmpty() || username.isEmpty() || password.isEmpty() || levelName.equals("- Pilih Level -")) {

JOptionPane.showMessageDialog(this, "Semua field harus diisi!", "Error", JOptionPane.ERROR\_MESSAGE);

return;

}

int idLevel = getIdLevelByName(levelName);

if (idLevel <= 0) { // Jika idLevel tidak valid (misal 0 atau -1 dari getIdLevelByName)

JOptionPane.showMessageDialog(this, "Level tidak valid atau tidak ditemukan di database.", "Error", JOptionPane.ERROR\_MESSAGE);

return;

}

try {

// Cek apakah username sudah ada

String checkSql = "SELECT COUNT(\*) FROM tbl\_user WHERE username = ? AND id\_user != ?";

PreparedStatement checkPs = koneksi.con.prepareStatement(checkSql);

checkPs.setString(1, username);

checkPs.setString(2, idUser); // Untuk kasus EDIT, agar tidak konflik dengan diri sendiri

ResultSet checkRs = checkPs.executeQuery();

if(checkRs.next() && checkRs.getInt(1) > 0){

JOptionPane.showMessageDialog(this, "Username '" + username + "' sudah digunakan!", "Error", JOptionPane.ERROR\_MESSAGE);

IUser.requestFocus();

return;

}

String sql = "INSERT INTO tbl\_user (id\_user, id\_level, nama\_user, jk, NOPE, username, password) VALUES (?, ?, ?, ?, ?, ?, ?)";

PreparedStatement ps = koneksi.con.prepareStatement(sql);

ps.setString(1, idUser);

ps.setInt(2, idLevel);

ps.setString(3, namaUser);

ps.setString(4, jk);

ps.setString(5, nope);

ps.setString(6, username);

ps.setString(7, password);

int rowsInserted = ps.executeUpdate();

if (rowsInserted > 0) {

JOptionPane.showMessageDialog(this, "Data Barang Berhasil Dimasukan"); // Sesuai gambar

load\_data();

bersih();

// IDOtomatis(); // Sudah dipanggil di dalam bersih()

}

} catch (SQLException e) {

// Cek duplikasi ID\_USER jika primary key violation

if (e.getSQLState().startsWith("23")){ // Kode error untuk integrity constraint violation

JOptionPane.showMessageDialog(this, "ID User '" + idUser + "' sudah ada atau terjadi kesalahan integritas data.\nDetail: " + e.getMessage(), "Error Simpan", JOptionPane.ERROR\_MESSAGE);

} else {

JOptionPane.showMessageDialog(this, "Error saat menyimpan data: " + e.getMessage(), "Error Simpan", JOptionPane.ERROR\_MESSAGE);

}

e.printStackTrace();

}

}//GEN-LAST:event\_BSimpanActionPerformed

private void BEditActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_BEditActionPerformed

// TODO: Implementasi fungsi Edit

JOptionPane.showMessageDialog(this, "Fungsi Edit belum diimplementasikan.");

}//GEN-LAST:event\_BEditActionPerformed

private void BHapusActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_BHapusActionPerformed

// TODO: Implementasi fungsi Hapus

JOptionPane.showMessageDialog(this, "Fungsi Hapus belum diimplementasikan.");

}//GEN-LAST:event\_BHapusActionPerformed

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(FUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(FUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(FUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(FUser.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new FUser().setVisible(true);

}

});

}

// Variables declaration - do not modify//GEN-BEGIN:variables

private javax.swing.JButton BEdit;

private javax.swing.JButton BHapus;

private javax.swing.JButton BSimpan;

private javax.swing.JComboBox<String> CLevel;

private javax.swing.ButtonGroup Gjk;

private javax.swing.JTextField IIdUser;

private javax.swing.JTextField INamaUser;

private javax.swing.JTextField INope;

private javax.swing.JPasswordField IPassword;

private javax.swing.JTextField IUser;

private javax.swing.JRadioButton JkL;

private javax.swing.JRadioButton JkP;

private javax.swing.JTable TUser;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel10;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabel4;

private javax.swing.JLabel jLabel5;

private javax.swing.JLabel jLabel6;

private javax.swing.JLabel jLabel7;

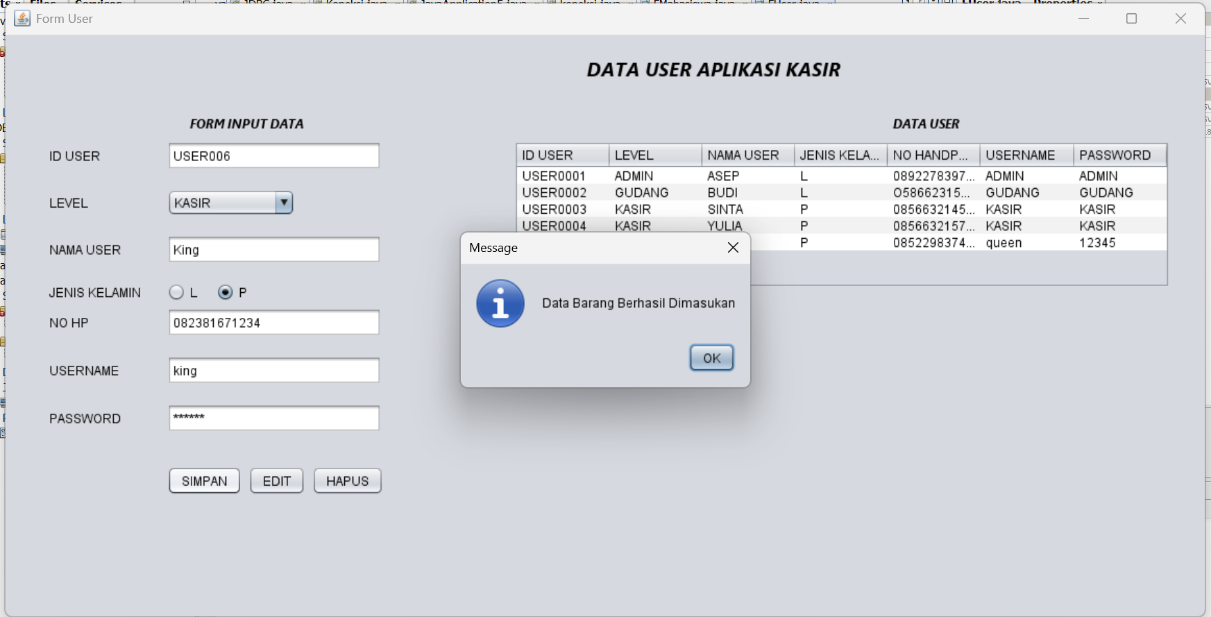
private javax.swing.JLabel jLabel8;

private javax.swing.JLabel jLabel9;

private javax.swing.JScrollPane jScrollPane1;

// End of variables declaration//GEN-END:variables

}



Analisis : Struktur basis data yang terdiri dari tabel tbl\_user dan tbl\_level mencerminkan hubungan satu ke banyak (one-to-many), di mana setiap level dalam tbl\_level dapat dimiliki oleh banyak pengguna di tbl\_user, namun setiap pengguna hanya memiliki satu level. Tabel tbl\_level bertindak sebagai referensi atau lookup table untuk mendefinisikan jenis level pengguna (misalnya admin, user biasa, dsb.), sedangkan tabel tbl\_user menyimpan data pribadi pengguna seperti nama, jenis kelamin, nomor telepon, username, dan password. Penggunaan ID\_LEVEL sebagai foreign key dalam tbl\_user menjamin integritas referensial antara kedua tabel. Selain itu, pemilihan tipe data VARCHAR untuk sebagian besar kolom memberikan fleksibilitas penyimpanan teks, sementara kolom kunci seperti ID\_USER dan ID\_LEVEL ditandai sebagai NOT NULL dan PRIMARY KEY, yang menegaskan pentingnya nilai unik dan tidak boleh kosong dalam menjaga keunikan data pengguna dan level. Struktur ini cukup umum dan efisien digunakan dalam sistem manajemen pengguna seperti aplikasi web atau sistem informasi internal.

# BAB III

# POSTTEST

## **1. Jelaskan Package dibawah ini :**

import java.sql.ResultSet;

Package: java.sql

Kelas/Interface: ResultSet (Interface)

Penjelasan: Package java.sql berisi API (Application Programming Interface) untuk JDBC (Java Database Connectivity), yang memungkinkan program Java berinteraksi dengan berbagai macam database. ResultSet adalah sebuah *interface* yang merepresentasikan hasil dari eksekusi query database (biasanya query SELECT). Objek ResultSet menyimpan data yang dikembalikan dalam bentuk tabel, di mana kita bisa menavigasi baris demi baris dan mengambil nilai dari setiap kolom. Kita bisa menganggapnya sebagai tabel data sementara di memori yang berisi hasil dari query.

import java.sql.Statement;

Package: java.sql

Kelas/Interface: Statement (Interface)

Penjelasan: Masih dari package java.sql, Statement adalah *interface* yang digunakan untuk mengeksekusi perintah SQL statis (perintah SQL yang tidak memiliki parameter input dinamis) dan mengembalikan hasil yang dihasilkannya. Setelah koneksi ke database berhasil dibuat, objek Statement dibuat dari objek Connection. Melalui objek Statement inilah kita bisa mengirimkan query SQL seperti SELECT, INSERT, UPDATE, atau DELETE ke database.

import javax.swing.JOptionPane;

Package: javax.swing

Kelas/Interface: JOptionPane (Kelas)

Penjelasan: Package javax.swing menyediakan kelas-kelas untuk membuat antarmuka pengguna grafis (GUI) dengan tampilan dan nuansa (look and feel) yang modern (Swing). JOptionPane adalah sebuah kelas yang memudahkan kita untuk menampilkan berbagai macam dialog box standar kepada pengguna, seperti dialog pesan informasi, dialog peringatan (warning), dialog error, dialog konfirmasi (yes/no/cancel), dan dialog input sederhana. Ini sangat berguna untuk interaksi cepat dengan pengguna tanpa perlu membuat frame atau dialog custom yang kompleks.

import javax.swing.table.DefaultTableModel;

Package: javax.swing.table

Kelas/Interface: DefaultTableModel (Kelas)

Penjelasan: Package javax.swing.table adalah bagian dari Swing yang khusus menangani komponen tabel (JTable). DefaultTableModel adalah implementasi konkret dari interface TableModel. Kelas ini menyediakan cara yang paling umum dan mudah untuk mengelola data yang akan ditampilkan dalam sebuah JTable. Kita bisa menggunakannya untuk menentukan nama kolom, jumlah baris, dan data di setiap sel tabel, serta untuk menambah, menghapus, atau memodifikasi baris data secara dinamis.

# BAB IV

# TUGAS

## Buatlah FData

/\*

 \* To change this license header, choose License Headers in Project Properties.

 \* To change this template file, choose Tools | Templates

 \* and open the template in the editor.

 \*/

package javaapplication5;

/\*\*

 \*

 \* @author Muhammad Rizal Nur F

 \*/

import javax.swing.table.DefaultTableModel;

import javax.swing.JOptionPane;

import java.sql.Statement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.PreparedStatement;

import java.sql.Connection; // Import Connection

public class FMahasiswa extends javax.swing.JFrame {

    Statement st; // Lebih baik membuat statement baru setiap kali dibutuhkan atau gunakan PreparedStatement

    ResultSet rs; // Sama seperti statement

    koneksi koneksi; // Nama kelas 'Koneksi' dengan K besar

    DefaultTableModel modelTabelMahasiswa;

    Connection connDB; // Variabel untuk menyimpan koneksi yang aktif

    public FMahasiswa() {

        koneksi = new koneksi(); // Instansiasi kelas Koneksi

        initComponents(); // Inisialisasi komponen GUI dari NetBeans Designer

        this.connDB = koneksi.con; // Dapatkan objek koneksi

        if (this.connDB == null) {

            JOptionPane.showMessageDialog(this,

                "Tidak dapat terhubung ke database. Aplikasi mungkin tidak berfungsi dengan benar.",

                "Error Koneksi Kritis", JOptionPane.ERROR\_MESSAGE);

            // Menonaktifkan tombol-tombol jika koneksi gagal

            BSimpan.setEnabled(false);

            BHapus.setEnabled(false);

            BBersih.setEnabled(false);

        } else {

            System.out.println("Koneksi database siap digunakan di FMahasiswa.");

            setupTable();

            load\_data();

            load\_prodi();

            bersih();

        }

        setLocationRelativeTo(null); // Form di tengah layar

    }

    private void setupTable() {

        Object header[] = {"NIM", "NAMA", "JENIS KELAMIN", "PRODI", "NO HP", "ALAMAT"};

        modelTabelMahasiswa = new DefaultTableModel(null, header) {

            @Override

            public boolean isCellEditable(int row, int column) {

                return false; // Membuat sel tabel tidak bisa diedit langsung

            }

        };

        TabelMahasiswa.setModel(modelTabelMahasiswa);

    }

    private void load\_data() {

        if (connDB == null) return; // Jangan lanjutkan jika tidak ada koneksi

        modelTabelMahasiswa.setRowCount(0);

        String sql = "SELECT nim, nama, jenis\_kelamin, prodi, no\_hp, alamat FROM tbl\_mahasiswa ORDER BY nim ASC"; // Tambahkan ORDER BY

        try (Statement st = connDB.createStatement(); // try-with-resources untuk Statement

             ResultSet rs = st.executeQuery(sql)) {   // try-with-resources untuk ResultSet

            while (rs.next()) {

                String nim = rs.getString("nim");

                String nama = rs.getString("nama");

                String jk = rs.getString("jenis\_kelamin");

                String prodi = rs.getString("prodi");

                String nohp = rs.getString("no\_hp");

                String alamat = rs.getString("alamat");

                String data[] = {nim, nama, jk, prodi, nohp, alamat};

                modelTabelMahasiswa.addRow(data);

            }

        } catch (SQLException e) {

            JOptionPane.showMessageDialog(this, "Gagal memuat data mahasiswa: " + e.getMessage(), "Error Database", JOptionPane.ERROR\_MESSAGE);

            e.printStackTrace();

        }

    }

    private void load\_prodi() {

        CProdi.removeAllItems();

        CProdi.addItem("Pilih Prodi"); // Item default

        CProdi.addItem("Teknik Informatika");

        CProdi.addItem("Sistem Informasi");

        CProdi.addItem("Manajemen Informatika");

        CProdi.addItem("Desain Komunikasi Visual");

        CProdi.addItem("Teknik Sipil");

        CProdi.addItem("Akuntansi");

        // Tambahkan prodi lain jika perlu

    }

    private void bersih() {

        INim.setText("");

        INama.setText("");

        GJenisKelamin.clearSelection();

        RBLaki.setSelected(true); // Default Laki-laki (atau sesuaikan)

        CProdi.setSelectedIndex(0); // Pilih "Pilih Prodi"

        INoHp.setText("");

        IAlamat.setText("");

        INim.setEnabled(true); // Atau setEditable(true)

        BSimpan.setText("SIMPAN");

        BHapus.setEnabled(false);

        INim.requestFocus();

        TabelMahasiswa.clearSelection(); // Hapus seleksi pada tabel

    }

    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents

    private void initComponents() {

        GJenisKelamin = new javax.swing.ButtonGroup();

        jLabel9 = new javax.swing.JLabel();

        jPanel1 = new javax.swing.JPanel();

        jLabel1 = new javax.swing.JLabel();

        jLabel2 = new javax.swing.JLabel();

        INim = new javax.swing.JTextField();

        jLabel3 = new javax.swing.JLabel();

        INama = new javax.swing.JTextField();

        jLabel4 = new javax.swing.JLabel();

        RBLaki = new javax.swing.JRadioButton();

        RBPerempuan = new javax.swing.JRadioButton();

        jLabel5 = new javax.swing.JLabel();

        CProdi = new javax.swing.JComboBox<>();

        jLabel6 = new javax.swing.JLabel();

        INoHp = new javax.swing.JTextField();

        jLabel7 = new javax.swing.JLabel();

        jScrollPane2 = new javax.swing.JScrollPane();

        IAlamat = new javax.swing.JTextArea();

        BSimpan = new javax.swing.JButton();

        BHapus = new javax.swing.JButton();

        BBersih = new javax.swing.JButton();

        jPanel2 = new javax.swing.JPanel();

        jLabel10 = new javax.swing.JLabel();

        jScrollPane1 = new javax.swing.JScrollPane();

        TabelMahasiswa = new javax.swing.JTable();

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

        setTitle("Form Data Mahasiswa");

        jLabel9.setFont(new java.awt.Font("Segoe UI", 3, 18)); // NOI18N

        jLabel9.setHorizontalAlignment(javax.swing.SwingConstants.CENTER);

        jLabel9.setText("DATA MAHASISWA");

        jPanel1.setBorder(javax.swing.BorderFactory.createEtchedBorder());

        jLabel1.setFont(new java.awt.Font("Segoe UI", 1, 12)); // NOI18N

        jLabel1.setText("FORM INPUT DATA MAHASISWA");

        jLabel2.setText("NIM");

        jLabel3.setText("Nama Mahasiswa");

        jLabel4.setText("Jenis Kelamin");

        GJenisKelamin.add(RBLaki);

        RBLaki.setText("Laki-laki");

        GJenisKelamin.add(RBPerempuan);

        RBPerempuan.setText("Perempuan");

        jLabel5.setText("Program Studi");

        CProdi.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "Pilih Prodi" }));

        jLabel6.setText("No. HP");

        jLabel7.setText("Alamat");

        IAlamat.setColumns(20);

        IAlamat.setRows(5);

        jScrollPane2.setViewportView(IAlamat);

        BSimpan.setText("SIMPAN");

        BSimpan.addActionListener(new java.awt.event.ActionListener() {

            public void actionPerformed(java.awt.event.ActionEvent evt) {

                BSimpanActionPerformed(evt);

            }

        });

        BHapus.setText("HAPUS");

        BHapus.setEnabled(false);

        BHapus.addActionListener(new java.awt.event.ActionListener() {

            public void actionPerformed(java.awt.event.ActionEvent evt) {

                BHapusActionPerformed(evt);

            }

        });

        BBersih.setText("BERSIH");

        BBersih.addActionListener(new java.awt.event.ActionListener() {

            public void actionPerformed(java.awt.event.ActionEvent evt) {

                BBersihActionPerformed(evt);

            }

        });

        javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

        jPanel1.setLayout(jPanel1Layout);

        jPanel1Layout.setHorizontalGroup(

            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addGroup(jPanel1Layout.createSequentialGroup()

                .addContainerGap()

                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                    .addComponent(jLabel1)

                    .addGroup(jPanel1Layout.createSequentialGroup()

                        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                            .addComponent(jLabel3)

                            .addComponent(jLabel2)

                            .addComponent(jLabel4)

                            .addComponent(jLabel5)

                            .addComponent(jLabel6)

                            .addComponent(jLabel7))

                        .addGap(18, 18, 18)

                        .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

                            .addGroup(jPanel1Layout.createSequentialGroup()

                                .addComponent(BSimpan)

                                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                                .addComponent(BHapus)

                                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                                .addComponent(BBersih))

                            .addComponent(INim)

                            .addComponent(INama)

                            .addGroup(jPanel1Layout.createSequentialGroup()

                                .addComponent(RBLaki)

                                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                                .addComponent(RBPerempuan))

                            .addComponent(CProdi, 0, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

                            .addComponent(INoHp)

                            .addComponent(jScrollPane2, javax.swing.GroupLayout.DEFAULT\_SIZE, 250, Short.MAX\_VALUE))))

                .addContainerGap(19, Short.MAX\_VALUE))

        );

        jPanel1Layout.setVerticalGroup(

            jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addGroup(jPanel1Layout.createSequentialGroup()

                .addContainerGap()

                .addComponent(jLabel1)

                .addGap(18, 18, 18)

                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(jLabel2)

                    .addComponent(INim, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(jLabel3)

                    .addComponent(INama, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(jLabel4)

                    .addComponent(RBLaki)

                    .addComponent(RBPerempuan))

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(jLabel5)

                    .addComponent(CProdi, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(jLabel6)

                    .addComponent(INoHp, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                    .addComponent(jLabel7)

                    .addComponent(jScrollPane2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE))

                .addGap(18, 18, 18)

                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)

                    .addComponent(BSimpan)

                    .addComponent(BHapus)

                    .addComponent(BBersih))

                .addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

        );

        jPanel2.setBorder(javax.swing.BorderFactory.createEtchedBorder());

        jLabel10.setFont(new java.awt.Font("Segoe UI", 1, 12)); // NOI18N

        jLabel10.setText("TABEL DATA MAHASISWA");

        TabelMahasiswa.setModel(new javax.swing.table.DefaultTableModel(

            new Object [][] {

                {null, null, null, null, null, null},

                {null, null, null, null, null, null},

                {null, null, null, null, null, null},

                {null, null, null, null, null, null}

            },

            new String [] {

                "NIM", "Nama", "Jenis Kelamin", "Prodi", "No HP", "Alamat"

            }

        ));

        TabelMahasiswa.addMouseListener(new java.awt.event.MouseAdapter() {

            public void mouseClicked(java.awt.event.MouseEvent evt) {

                TabelMahasiswaMouseClicked(evt);

            }

        });

        jScrollPane1.setViewportView(TabelMahasiswa);

        javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

        jPanel2.setLayout(jPanel2Layout);

        jPanel2Layout.setHorizontalGroup(

            jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addGroup(jPanel2Layout.createSequentialGroup()

                .addContainerGap()

                .addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                    .addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 566, Short.MAX\_VALUE)

                    .addGroup(jPanel2Layout.createSequentialGroup()

                        .addComponent(jLabel10)

                        .addGap(0, 0, Short.MAX\_VALUE)))

                .addContainerGap())

        );

        jPanel2Layout.setVerticalGroup(

            jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addGroup(jPanel2Layout.createSequentialGroup()

                .addContainerGap()

                .addComponent(jLabel10)

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                .addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 0, Short.MAX\_VALUE)

                .addContainerGap())

        );

        javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

        getContentPane().setLayout(layout);

        layout.setHorizontalGroup(

            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addGroup(layout.createSequentialGroup()

                .addContainerGap()

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

                    .addComponent(jLabel9, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

                    .addGroup(layout.createSequentialGroup()

                        .addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

                        .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

                        .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)))

                .addContainerGap())

        );

        layout.setVerticalGroup(

            layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

            .addGroup(layout.createSequentialGroup()

                .addContainerGap()

                .addComponent(jLabel9)

                .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

                .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

                    .addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

                    .addComponent(jPanel2, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

                .addContainerGap(28, Short.MAX\_VALUE))

        );

        pack();

    }// </editor-fold>//GEN-END:initComponents

    private void BBersihActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_BBersihActionPerformed

        bersih();

    }//GEN-LAST:event\_BBersihActionPerformed

    private void BSimpanActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_BSimpanActionPerformed

        if (connDB == null) {

            JOptionPane.showMessageDialog(this, "Koneksi database tidak tersedia.", "Error", JOptionPane.ERROR\_MESSAGE);

            return;

        }

        String nim = INim.getText().trim();

        String nama = INama.getText().trim();

        String jenisKelamin = "";

        if (RBLaki.isSelected()) {

            jenisKelamin = "Laki-laki";

        } else if (RBPerempuan.isSelected()) {

            jenisKelamin = "Perempuan";

        }

        // Ambil nilai prodi, pastikan tidak null

        String prodi = CProdi.getSelectedItem() != null ? CProdi.getSelectedItem().toString() : "";

        String noHp = INoHp.getText().trim();

        String alamat = IAlamat.getText().trim();

        if (nim.isEmpty() || nama.isEmpty() || jenisKelamin.isEmpty() || prodi.equals("Pilih Prodi") || prodi.isEmpty()) {

            JOptionPane.showMessageDialog(this, "NIM, Nama, Jenis Kelamin, dan Prodi wajib diisi/dipilih!", "Peringatan", JOptionPane.WARNING\_MESSAGE);

            return;

        }

        String sql;

        boolean isUpdate = BSimpan.getText().equals("UPDATE");

        try {

            PreparedStatement pstmt;

            if (isUpdate) { // Mode Update

                sql = "UPDATE tbl\_mahasiswa SET nama=?, jenis\_kelamin=?, prodi=?, no\_hp=?, alamat=? WHERE nim=?";

                pstmt = connDB.prepareStatement(sql);

                pstmt.setString(1, nama);

                pstmt.setString(2, jenisKelamin);

                pstmt.setString(3, prodi);

                pstmt.setString(4, noHp);

                pstmt.setString(5, alamat);

                pstmt.setString(6, nim); // NIM untuk klausa WHERE

            } else { // Mode Simpan

                 // Cek dulu apakah NIM sudah ada

                try (Statement stmtCek = connDB.createStatement();

                     ResultSet rsCek = stmtCek.executeQuery("SELECT nim FROM tbl\_mahasiswa WHERE nim='" + nim + "'")) {

                    if (rsCek.next()) {

                        JOptionPane.showMessageDialog(this, "NIM " + nim + " sudah terdaftar!", "Error Simpan", JOptionPane.ERROR\_MESSAGE);

                        return; // Hentikan proses simpan jika NIM sudah ada

                    }

                } // stmtCek dan rsCek otomatis tertutup di sini

                sql = "INSERT INTO tbl\_mahasiswa (nim, nama, jenis\_kelamin, prodi, no\_hp, alamat) VALUES (?, ?, ?, ?, ?, ?)";

                pstmt = connDB.prepareStatement(sql);

                pstmt.setString(1, nim);

                pstmt.setString(2, nama);

                pstmt.setString(3, jenisKelamin);

                pstmt.setString(4, prodi);

                pstmt.setString(5, noHp);

                pstmt.setString(6, alamat);

            }

            int affectedRows = pstmt.executeUpdate();

            if (affectedRows > 0) {

                JOptionPane.showMessageDialog(this, "Data berhasil " + (isUpdate ? "diupdate!" : "disimpan!"), "Sukses", JOptionPane.INFORMATION\_MESSAGE);

                load\_data();

                bersih();

            } else {

                 JOptionPane.showMessageDialog(this, "Data gagal " + (isUpdate ? "diupdate (NIM tidak ditemukan?)." : "disimpan."), "Gagal", JOptionPane.ERROR\_MESSAGE);

            }

            pstmt.close(); // Tutup PreparedStatement

        } catch (SQLException e) {

            JOptionPane.showMessageDialog(this, "Gagal " + (isUpdate ? "mengupdate" : "menyimpan") + " data: " + e.getMessage(), "Error Database", JOptionPane.ERROR\_MESSAGE);

            e.printStackTrace();

        }

    }//GEN-LAST:event\_BSimpanActionPerformed

    private void TabelMahasiswaMouseClicked(java.awt.event.MouseEvent evt) {//GEN-FIRST:event\_TabelMahasiswaMouseClicked

        int baris = TabelMahasiswa.getSelectedRow();

        if (baris != -1) { // Pastikan ada baris yang dipilih

            INim.setText(modelTabelMahasiswa.getValueAt(baris, 0).toString());

            INama.setText(modelTabelMahasiswa.getValueAt(baris, 1).toString());

            String jk = modelTabelMahasiswa.getValueAt(baris, 2).toString();

            if (jk.equalsIgnoreCase("Laki-laki")) { // Gunakan equalsIgnoreCase untuk perbandingan string

                RBLaki.setSelected(true);

            } else if (jk.equalsIgnoreCase("Perempuan")) {

                RBPerempuan.setSelected(true);

            } else {

                GJenisKelamin.clearSelection(); // Jika tidak L atau P, bersihkan pilihan

            }

            // Hati-hati jika item prodi di tabel tidak ada di ComboBox

            String prodiDiTabel = modelTabelMahasiswa.getValueAt(baris, 3).toString();

            boolean prodiDitemukan = false;

            for (int i = 0; i < CProdi.getItemCount(); i++) {

                if (CProdi.getItemAt(i).equalsIgnoreCase(prodiDiTabel)) {

                    CProdi.setSelectedIndex(i);

                    prodiDitemukan = true;

                    break;

                }

            }

            if (!prodiDitemukan && !prodiDiTabel.isEmpty()) {

                 // Jika prodi dari tabel tidak ada di ComboBox, bisa tambahkan ke ComboBox (opsional)

                 // CProdi.addItem(prodiDiTabel);

                 // CProdi.setSelectedItem(prodiDiTabel);

                 // Atau set ke default "Pilih Prodi" jika tidak ingin menambah

                 CProdi.setSelectedIndex(0); // Atau tampilkan pesan

                 JOptionPane.showMessageDialog(this, "Program studi '" + prodiDiTabel + "' tidak ditemukan dalam pilihan.", "Info", JOptionPane.INFORMATION\_MESSAGE);

            }

            Object noHpObj = modelTabelMahasiswa.getValueAt(baris, 4);

            INoHp.setText(noHpObj != null ? noHpObj.toString() : "");

            Object alamatObj = modelTabelMahasiswa.getValueAt(baris, 5);

            IAlamat.setText(alamatObj != null ? alamatObj.toString() : "");

            INim.setEnabled(false); // Atau setEditable(false)

            BSimpan.setText("UPDATE");

            BHapus.setEnabled(true);

        }

    }//GEN-LAST:event\_TabelMahasiswaMouseClicked

    private void BHapusActionPerformed(java.awt.event.ActionEvent evt) {//GEN-FIRST:event\_BHapusActionPerformed

        if (connDB == null) {

             JOptionPane.showMessageDialog(this, "Koneksi database tidak tersedia.", "Error", JOptionPane.ERROR\_MESSAGE);

            return;

        }

        String nim = INim.getText();

        if (nim.isEmpty() || !BHapus.isEnabled()) { // Cek juga apakah tombol hapus aktif

            JOptionPane.showMessageDialog(this, "Pilih data dari tabel yang akan dihapus!", "Peringatan", JOptionPane.WARNING\_MESSAGE);

            return;

        }

        int konfirmasi = JOptionPane.showConfirmDialog(this,

                "Anda yakin ingin menghapus data mahasiswa dengan NIM '" + nim + "'?",

                "Konfirmasi Hapus", JOptionPane.YES\_NO\_OPTION, JOptionPane.WARNING\_MESSAGE);

        if (konfirmasi == JOptionPane.YES\_OPTION) {

            String sql = "DELETE FROM tbl\_mahasiswa WHERE nim = ?";

            try (PreparedStatement pstmt = connDB.prepareStatement(sql)) { // try-with-resources

                pstmt.setString(1, nim);

                int affectedRows = pstmt.executeUpdate();

                if (affectedRows > 0) {

                    JOptionPane.showMessageDialog(this, "Data berhasil dihapus!", "Sukses", JOptionPane.INFORMATION\_MESSAGE);

                    load\_data();

                    bersih();

                } else {

                    JOptionPane.showMessageDialog(this, "Data gagal dihapus (NIM tidak ditemukan).", "Gagal", JOptionPane.ERROR\_MESSAGE);

                }

            } catch (SQLException e) {

                JOptionPane.showMessageDialog(this, "Gagal menghapus data: " + e.getMessage(), "Error Database", JOptionPane.ERROR\_MESSAGE);

                e.printStackTrace();

            }

        }

    }//GEN-LAST:event\_BHapusActionPerformed

    public static void main(String args[]) {

        try {

            for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

                if ("Nimbus".equals(info.getName())) {

                    javax.swing.UIManager.setLookAndFeel(info.getClassName());

                    break;

                }

            }

        } catch (ClassNotFoundException | InstantiationException | IllegalAccessException | javax.swing.UnsupportedLookAndFeelException ex) {

            java.util.logging.Logger.getLogger(FMahasiswa.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

        }

        java.awt.EventQueue.invokeLater(new Runnable() {

            public void run() {

                new FMahasiswa().setVisible(true);

            }

        });

    }

    // Variables declaration - do not modify//GEN-BEGIN:variables

    private javax.swing.JButton BBersih;

    private javax.swing.JButton BHapus;

    private javax.swing.JButton BSimpan;

    private javax.swing.JComboBox<String> CProdi;

    private javax.swing.ButtonGroup GJenisKelamin;

    private javax.swing.JTextArea IAlamat;

    private javax.swing.JTextField INama;

    private javax.swing.JTextField INim;

    private javax.swing.JTextField INoHp;

    private javax.swing.JRadioButton RBLaki;

    private javax.swing.JRadioButton RBPerempuan;

    private javax.swing.JTable TabelMahasiswa;

    private javax.swing.JLabel jLabel1;

    private javax.swing.JLabel jLabel10;

    private javax.swing.JLabel jLabel2;

    private javax.swing.JLabel jLabel3;

    private javax.swing.JLabel jLabel4;

    private javax.swing.JLabel jLabel5;

    private javax.swing.JLabel jLabel6;

    private javax.swing.JLabel jLabel7;

    private javax.swing.JLabel jLabel9;

    private javax.swing.JPanel jPanel1;

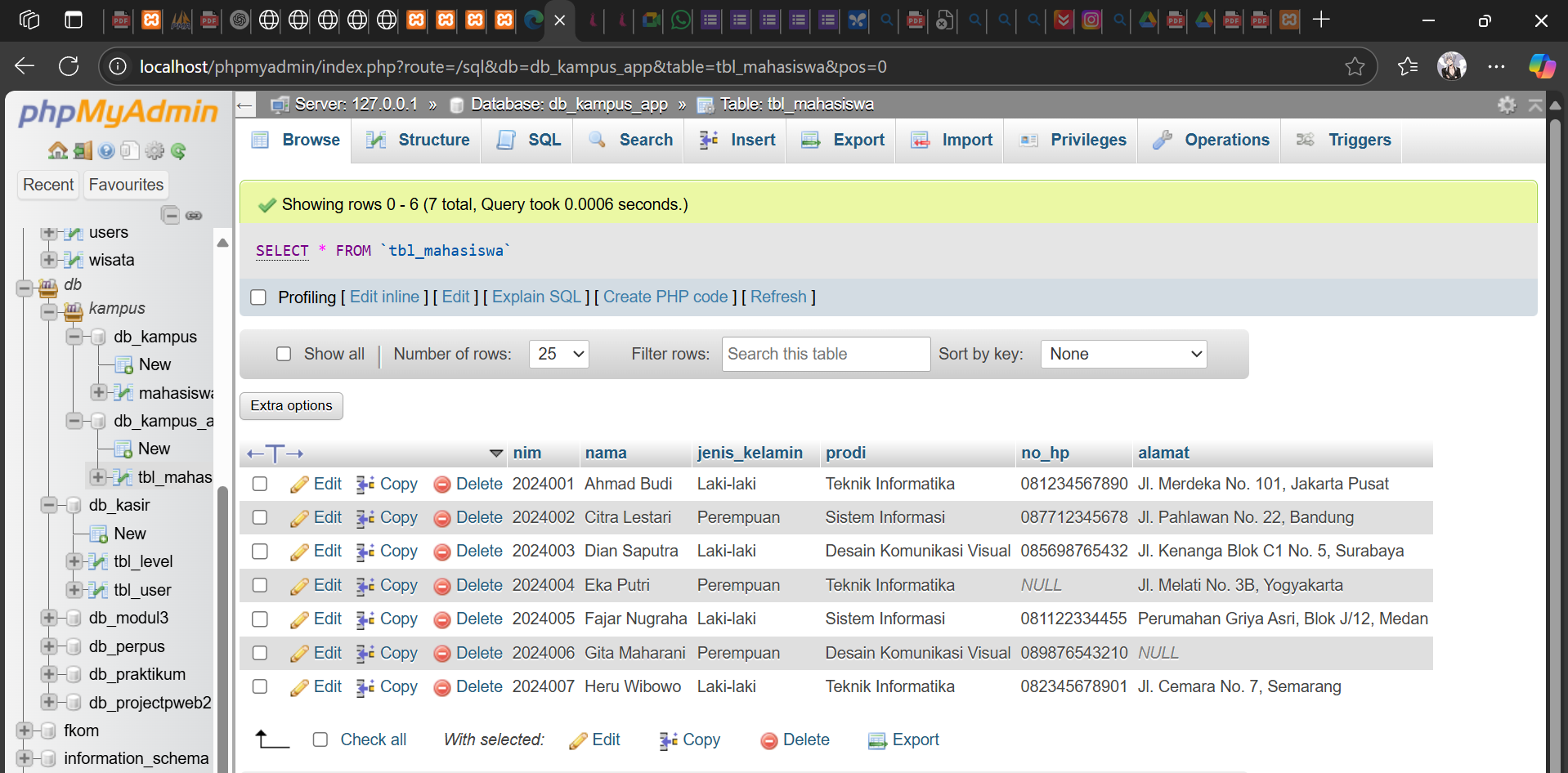
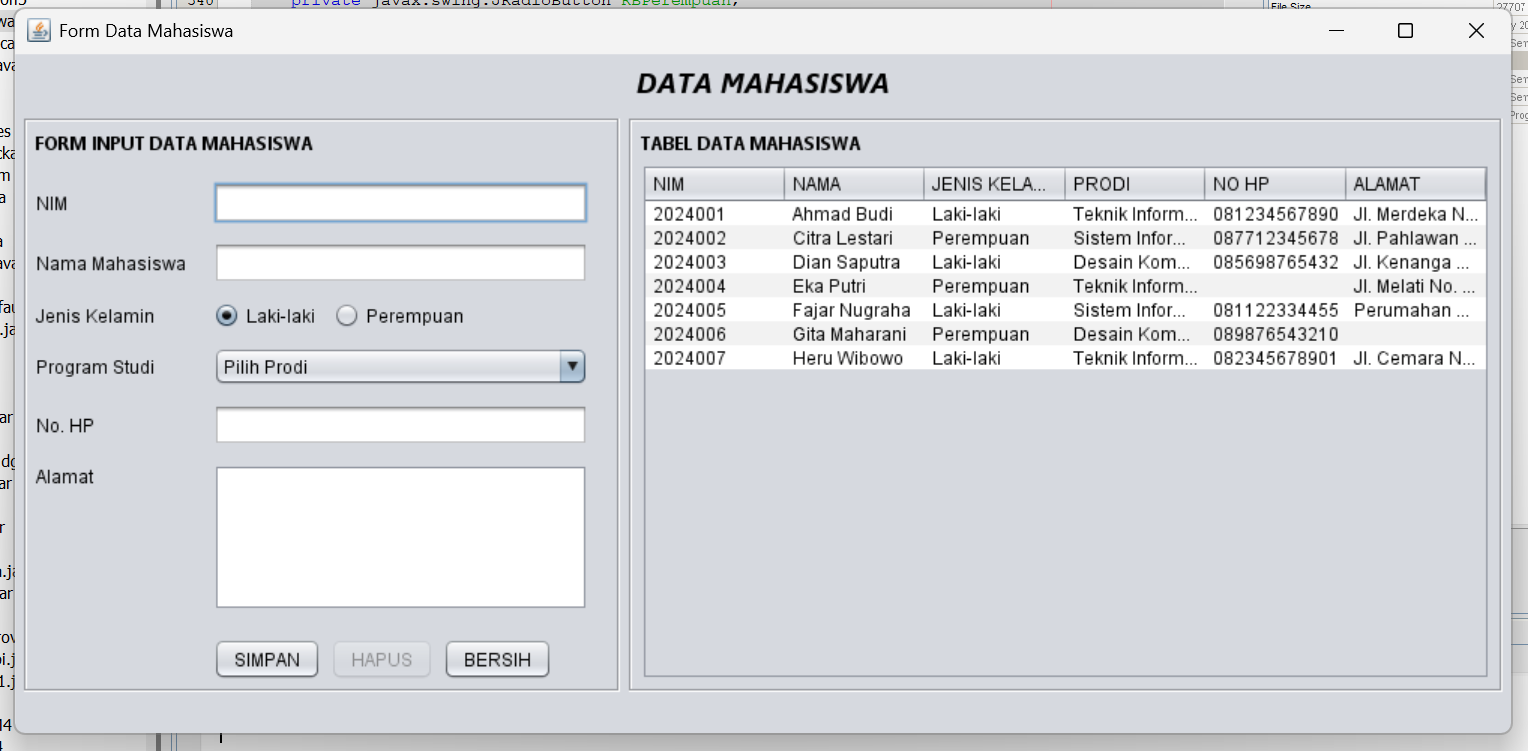
    private javax.swing.JPanel jPanel2;

    private javax.swing.JScrollPane jScrollPane1;

    private javax.swing.JScrollPane jScrollPane2;

    // End of variables declaration//GEN-END:variables

}

  
  
Analisis : Program FMahasiswa.java ini merupakan aplikasi desktop Java Swing yang dirancang untuk mengelola data mahasiswa (Create, Read, Update, Delete - CRUD) yang tersimpan dalam database MySQL. Aplikasi ini memiliki antarmuka pengguna grafis (GUI) yang terdiri dari form input data di sisi kiri (untuk NIM, nama, jenis kelamin, prodi, no HP, dan alamat) beserta tombol-tombol aksi (SIMPAN/UPDATE, HAPUS, BERSIH), dan sebuah tabel di sisi kanan untuk menampilkan daftar mahasiswa yang sudah ada. Interaksi utama pengguna adalah mengisi form untuk menambah data baru, memilih data dari tabel untuk diedit atau dihapus, dan membersihkan form.

Secara fungsional, program ini berhasil melakukan koneksi ke database db\_kampus\_app melalui kelas koneksi.java yang telah dikonfigurasi dengan benar. Metode load\_data() bertanggung jawab untuk mengambil semua data dari tabel tbl\_mahasiswa dan menampilkannya di JTable. Aksi simpan dan update menggunakan PreparedStatement untuk keamanan dan efisiensi, dengan logika untuk membedakan apakah operasi tersebut adalah penambahan data baru atau pembaruan data yang sudah ada (berdasarkan teks tombol "SIMPAN" atau "UPDATE" dan status edit NIM). Fungsi hapus juga diimplementasikan dengan konfirmasi pengguna, sementara fungsi bersih mereset form ke keadaan awal. Penanganan error dasar melalui JOptionPane juga disertakan untuk memberitahu pengguna jika terjadi masalah saat operasi database.

# BAB V

# KESIMPULAN

Dari sana, fokus beralih pada masalah yang lebih teknis di sisi backend, tepatnya dalam query SQL yang digunakan oleh aplikasi Java. Terjadi error SQLSyntaxErrorException, yang ternyata disebabkan oleh perbedaan nama kolom antara yang ada di kode program dan yang ada di database. Kode program menggunakan nohp, sementara tabel di database menggunakan NOPE. Setelah nama kolom disamakan, masalah ini pun terselesaikan.

Setelah itu, pengembangan aplikasi CRUD untuk data mahasiswa menjadi fokus utama. Tabel tbl\_mahasiswa dirancang dengan atribut seperti NIM, nama, jenis kelamin, prodi, nomor HP, dan alamat. Skrip SQL pun dibuat untuk membangun tabel dan mengisinya dengan data awal. Di sisi aplikasi, form GUI menggunakan Java Swing (melalui kelas FMahasiswa.java) dikembangkan, lengkap dengan input field dan tabel untuk menampilkan data. Kelas koneksi (koneksi.java) juga disiapkan untuk menjembatani aplikasi dengan database.

Dalam prosesnya, beberapa error muncul—yang cukup klasik dalam pengembangan berbasis JDBC. Salah satunya adalah error ClassNotFoundException, yang mengindikasikan bahwa driver JDBC belum dikenali. Ini biasanya karena library belum ditambahkan ke project, dan memang terbukti demikian. Setelah library mysql-connector-java disertakan ke dalam classpath NetBeans, koneksi pun mulai berhasil.

Namun tantangan belum selesai. Muncul error baru: tabel tbl\_mahasiswa tidak ditemukan di database. Setelah ditelusuri, rupanya ada kesalahan dalam konfigurasi URL koneksi—aplikasi ternyata tersambung ke database db\_kampus, sementara tabel yang dimaksud ada di db\_kampus\_app. Dengan menyesuaikan konfigurasi koneksi, masalah ini pun teratasi.

Akhirnya, aplikasi CRUD berjalan dengan baik. Data mahasiswa dapat ditampilkan, ditambah, diubah, dan dihapus melalui antarmuka yang dibangun dengan Swing. Semua fitur dasar berfungsi sebagaimana mestinya, dan proses debug yang dilalui menunjukkan pemahaman yang semakin matang terhadap integrasi antara Java, JDBC, dan MySQL, sekaligus memberikan pelajaran penting tentang ketelitian dalam pengelolaan koneksi dan struktur data.