

**Laporan Porjek UTS**  
**Pemrograman Komputer 2**



**Nama : Tri Joyo Priambodo**

**NIM : 18090099**

**Kelas : 4B**

**Prodi : D IV Teknik Informatika**

**Politeknik Harapan Bersama Tegal**

## A. Membuat database Dengan cara sbb

Server: 127.0.0.1 » Basis data: db\_joyo » Tabel: db\_login

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi Pelacakan Trigger

Nama tabel: db\_login    Tambah 1 column(s)    Kirim

Nama	Jenis	Panjang/Nilai	Bawaan	Penyortiran	Atribut	Tak Ternilai	Indeks
id	INT	50	Tidak ada			<input type="checkbox"/>	PRIMARY
username	VARCHAR	50	Tidak ada			<input type="checkbox"/>	---
password	VARCHAR	50	Tidak ada			<input type="checkbox"/>	---

Struktur

Server: 127.0.0.1 » Basis data: db\_joyo » Tabel: arsip

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi Pelacakan Trigger

Struktur tabel    Tampilan hubungan

#	Nama	Jenis	Penyortiran	Atribut	Tak Ternilai	Bawaan	Komentar	Ekstra	Tindakan
1	kode	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
2	nama	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
3	kategori	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
4	lokasi	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
5	deskripsi	varchar(50)	utf8mb4_general_ci		Tidak	Tidak ada			Ubah Hapus Lainnya
6	tanggal	date			Tidak	Tidak ada			Ubah Hapus Lainnya

Pilih Semua    Dengan pilihan:    Jelajahi    Ubah    Hapus    Utama    Unik    Indeks    Teks penuh    Add to central columns

Remove from central columns

Server: 127.0.0.1 » Basis data: db\_joyo » Tabel: db\_login

Jelajahi Struktur SQL Cari Tambahkan Ekspor Impor Hak Akses Operasi Pelacakan Trigger

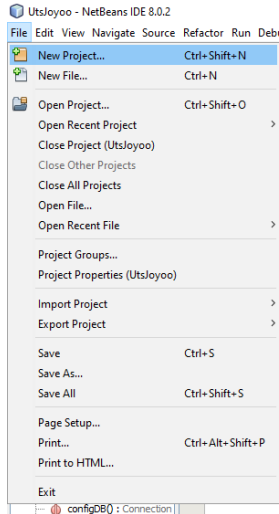
Kolom	Jenis	Fungsi	Tak Ternilai	Nilai
id	int(50)		<input type="checkbox"/>	1
username	varchar(50)		<input type="checkbox"/>	admin
password	varchar(50)	MD5	<input type="checkbox"/>	admin

Kirim

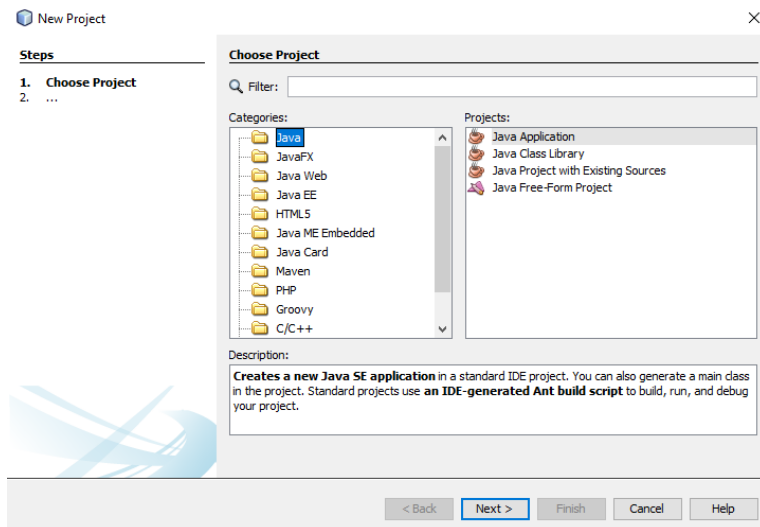
☒ Abaikan

## A. Membuat Project

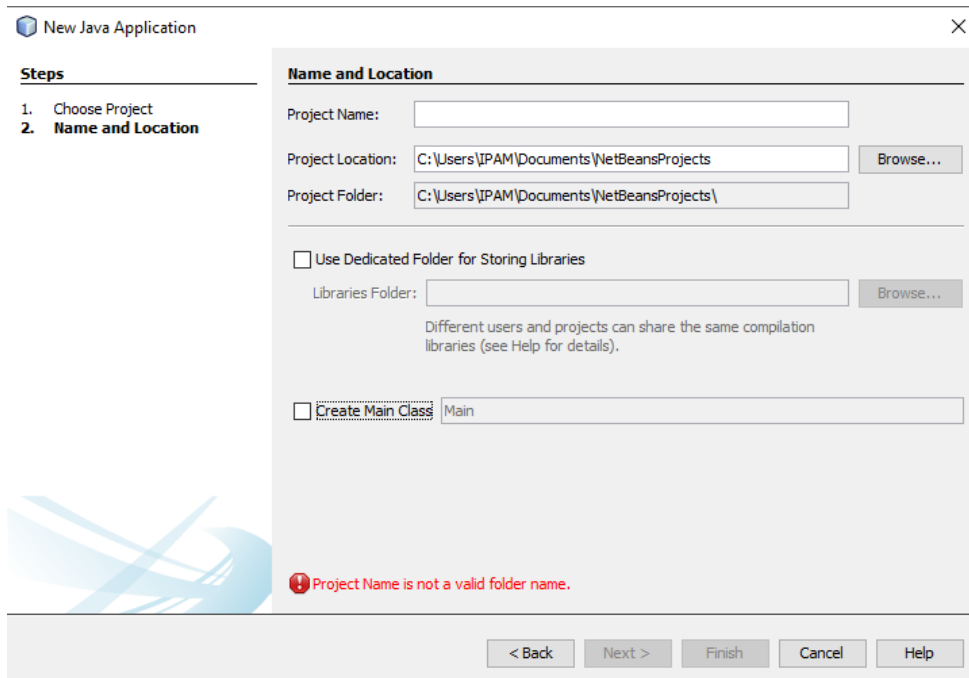
1. kita buka netbean
2. Lalu kita klik file -> new project



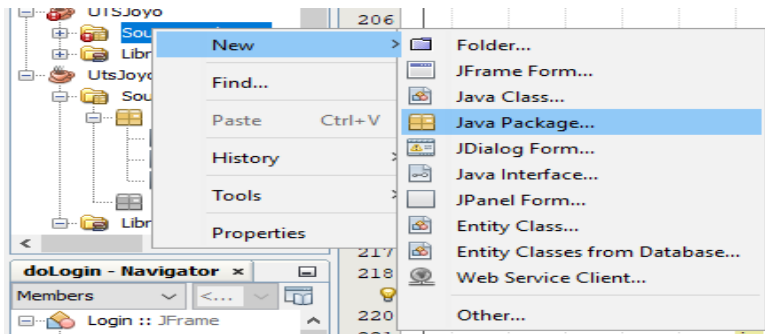
## 3. Lalu kita pilih java application



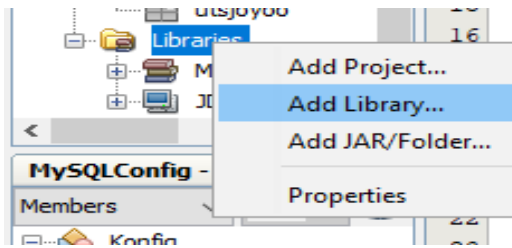
4. Setelah itu masukan nama project dan hapus ceklis di bawah



5. Setelah projek dibuat kita akan membuat package baru dengan cara klik kanan di source package, pilih new dan klik java package



6. Lalu dalam java package kita buat class Konfig dengan cara yang sama seperti diatas tapi kita pilih java class.
7. Sebelumnya jangan lupa kita masukan library MYSQL dengan cara add library.



8. Selanjutnya kita masuk lagi ke class konfigurasi dan menuliskan kode sbb.

```
package folder;

/*
 * 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.
 */

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
public class Konfig {

    /**
     * @author Asus
     */

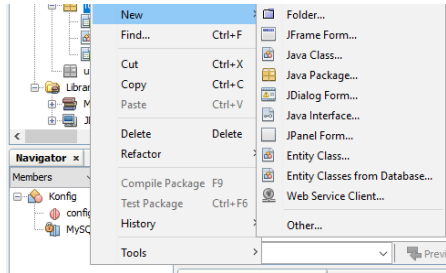
    private static Connection MySQLConfig;

    public static Connection configDB () throws SQLException{
        try {
```

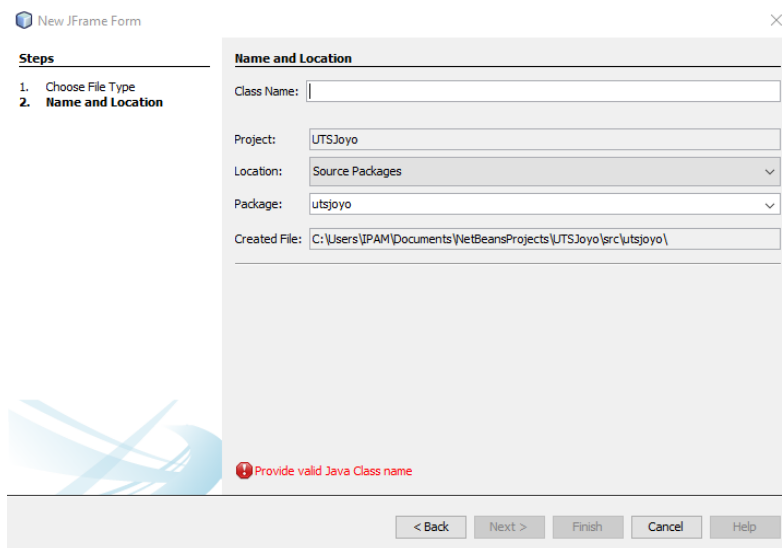
```
23
24         public static Connection configDB () throws SQLException{
25             try {
26                 String url = "jdbc:mysql://localhost:3306/db_joyo";
27                 String user = "root";
28                 String pass = "";
29
30                 DriverManager.registerDriver(new com.mysql.jdbc.Driver());
31                 MySQLConfig = DriverManager.getConnection(url , user , pass);
32
33             } catch (Exception e) {
34                 System.out.println("Koneksi Ke Database Gagal"+e.getMessage());
35             }
36             return MySQLConfig;
37         }
38     }
39 }
```

## Membuat Form Login

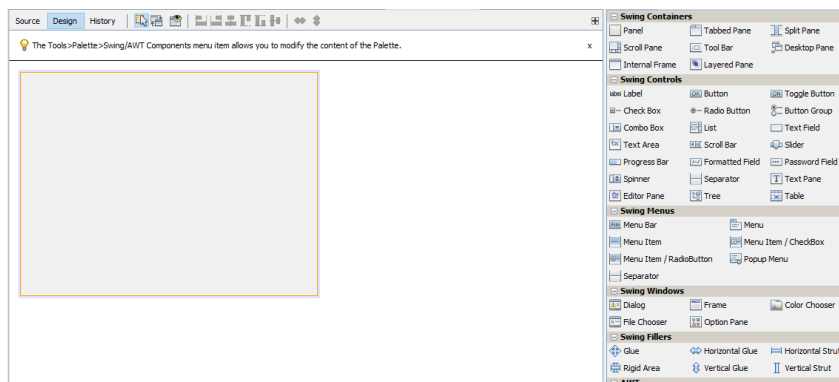
1. Setelah itu kita buat form login dengan cara klik kanan di package dan pilih jframe form



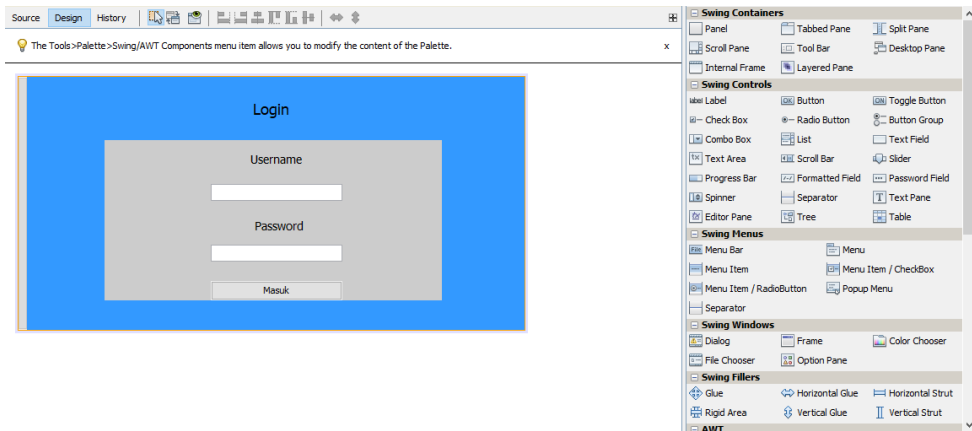
2. Lalu masukan nama dan klik finish



3. Tampilan awalnya adalah seperti ini



4. Lalu kita edit dengan memasukkan 1 text field,1 password field dan 3 label hingga seperti ini.



5. Kita ubah nama variable tiap field dan button  
6. Lalu kita klik source untuk memasukkan aksi/perintah di form login  
Kodenya adalah sbb

```
Source Design History
package folder;

/*
 * 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.
 */

import folder.Konfig;
import java.awt.Frame;
import java.security.MessageDigest;
import java.security.NoSuchAlgorithmException;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.Statement;
import javax.swing.JOptionPane;

/**
 *
 * @author Asus
 */
public class Login extends javax.swing.JFrame {

    /**
     *
     */
}
```

```
Source Design History
public class Login extends javax.swing.JFrame {

    /**
     * Creates new form Login
     */
    public Login() {
        initComponents();
    }

    /**
     * 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")
    Generated Code

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        Look and feel setting code (optional)
        //</editor-fold>
    }
}
```

```
Source Design History
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        new Login().setVisible(true);
    }
});
173
174
175
176
177
178 // Variables declaration - do not modify
179 private javax.swing.JButton btnlogin;
180 private javax.swing.JLabel jLabel1;
181 private javax.swing.JLabel jLabel2;
182 private javax.swing.JLabel jLabel3;
183 private javax.swing.JPanel jPanel1;
184 private javax.swing.JPanel jPanel2;
185 private javax.swing.JPasswordField txtpassword;
186 private javax.swing.JTextField txtusername;
187 // End of variables declaration
188 public void doLogin() {
189     btnlogin.setEnabled(false);
190     String user = txtusername.getText();
191     char[] pass = txtpassword.getPassword();
192     if (user.isEmpty()) {
193         JOptionPane.showMessageDialog(this, "Username harus diisi");
194         txtusername.requestFocus();
195     } else if (pass.length == 0) {
```

```
Source Design History
195     } else if (pass.length == 0) {
196         JOptionPane.showMessageDialog(this, "Password harus diisi");
197         txtpassword.requestFocus();
198     } else {
199         String password = new String(pass);
200         String md5_pass = md5(password);
201         try {
202             Connection c = Konfig.configDB();
203             Statement st = c.createStatement();
204             String q = " SELECT * FROM db_login "
205                 + "WHERE username = '" + user + "' "
206                 + "AND password = '" + md5_pass + "'";
207             ResultSet rs = st.executeQuery(q);
208             int n = 0;
209             while (rs.next()) {
210                 String u = rs.getString("username");
211                 String p = rs.getString("password");
212                 n++;
213             }
214             if (n == 0) {
215                 JOptionPane.showMessageDialog(this, "Username/password salah!");
216                 txtusername.requestFocus();
217                 btnlogin.setEnabled(true);
218             } else {
219                 Dokumen jy = new Dokumen();
```

```
Source Design History
217         btnlogin.setEnabled(true);
218     } else {
219         Dokumen jy = new Dokumen();
220         this.setVisible(false);
221         jy.setExtendedState(Frame.MAXIMIZED_BOTH);
222         jy.setVisible(true);
223     }
224 } catch (Exception e) {
225     System.out.println(e.getMessage());
226 }
227
228
229
230 public static String md5(String data) {
231     String hashed = "";
232     try {
233         MessageDigest md5 = MessageDigest.getInstance("MD5");
234         md5.update(data.getBytes());
235         byte[] byteData = md5.digest();
236         StringBuilder sb = new StringBuilder();
237         for (int i = 0; i < byteData.length; i++) {
238             sb.append(Integer.toString((byteData[i] & 0xff)
239                 + 0x100, 16).substring(1));
240         }
241         hashed = sb.toString();
242     } catch (NoSuchAlgorithmException e) {
```



```
Source Design History
229
230 public static String md5(String data) {
231     String hashed = "";
232     try {
233         MessageDigest md5 = MessageDigest.getInstance("MD5");
234         md5.update(data.getBytes());
235         byte[] byteData = md5.digest();
236         StringBuilder sb = new StringBuilder();
237         for (int i = 0; i < byteData.length; i++) {
238             sb.append(Integer.toString((byteData[i] & 0xff)
239                 + 0x100, 16).substring(1));
240         }
241         hashed = sb.toString();
242     } catch (NoSuchAlgorithmException e) {
243     }
244     return hashed;
245 }
246
247
```

## Membuat form Dokumen/Arsip

1. Kita buat JFrame form sama seperti diatas dengan hasil akhir seperti ini

Design Preview [Dokumen]

### Form Dokumen

Cari Nama

Kode Dokumen

Nama Dokumen

Kategori Dokumen

Lokasi Dokumen

Deskripsi Dokumen

Tanggal

Title 1	Title 2	Title 3	Title 4

2. Lalu jangan lupa ubah iap text field dan buttonya,dan ketikan kode sbb

```
Source Design History | [Icons]
import folder.Konfig;
14 import java.awt.Frame;
15 import java.awt.HeadlessException;
16 import java.sql.Connection;
17 import java.sql.SQLException;
18 import javax.swing.JOptionPane;
19 import javax.swing.table.DefaultTableModel;
20
21
22 public class Dokumen extends javax.swing.JFrame {
23
24     private void kosongkan_from() {
25         txtkodedokumen.setEditable(true);
26         txtkodedokumen.setText(null);
27         txtnamadokumen.setText(null);
28         cmbkatagoridokumen.setSelectedItem(this);
29         txtlokasi.setText(null);
30         txtdeskripsi.setText(null);
31         txttanggal.setText(null);
32     }
33
34     private void tampilkan_data() {
35         DefaultTableModel model = new DefaultTableModel();
36         model.addColumn("No.");
```

```
Source Design History | [Icons]
34 private void tampilkan_data() {
35     DefaultTableModel model = new DefaultTableModel();
36     model.addColumn("No.");
37     model.addColumn("Kode Dokumen");
38     model.addColumn("Nama Dokumen");
39     model.addColumn("Kategori Dokumen");
40     model.addColumn("Lokasi Dokumen");
41     model.addColumn("Deskripsi Dokumen");
42     model.addColumn("Tanggal");
43
44     String cari= txtcari.getText();
45     try {
46         int no = 1;
47         String sql = "SELECT *FROM tb_dokumen WHERE nama LIKE '%" +cari+"%' ";
48         java.sql.Connection conn = (Connection)Konfig.configDB();
49         java.sql.Statement stm = conn.createStatement();
50         java.sql.ResultSet res = stm.executeQuery(sql);
51
52         while (res.next()) {
53             model.addRow(new Object[]{no++,res.getString(1),
54                 res.getString(2),res.getString(3),res.getString(4),res.getString(5),res.getString(6)});
55         }
56         tabledokumen.setModel(model);
57     } catch (Exception e) {}
```

```

371 | // </editor-fold>
372 |
373 | private void btnTambahDataActionPerformed(java.awt.event.ActionEvent evt) {
374 |     try {
375 |         String sql = "INSERT INTO tb_dokumen VALUES ('"+txtkodedokumen.getText()+"','"+txtnamadokumen.getText()+"','"+cmbkatagoridokumen.getSelectedItem()+"','"+txtlokasi.get
376 |         java.sql.Connection conn = (Connection)Konfig.configDB();
377 |         java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
378 |         pstmt.execute();
379 |         JOptionPane.showMessageDialog(null, "Proses simpan Data berhasil");
380 |         tampilkan_data();
381 |         kosongkan_from();
382 |     } catch (HeadlessException | SQLException e) {
383 |         JOptionPane.showMessageDialog(this, e.getMessage());
384 |     }
385 |     txtkodedokumen.setFocusable(true);
386 | }
387 |
388 |
389 | private void btnSimpanActionPerformed(java.awt.event.ActionEvent evt) {
390 |     try {
391 |         String sql = "INSERT INTO tb_dokumen VALUES ('"+txtkodedokumen.getText()+"','"+txtnamadokumen.getText()+"','"+cmbkatagoridokumen.getSelectedItem()+"','"+txtlokasi.get
392 |         java.sql.Connection conn = (Connection)Konfig.configDB();
393 |         java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
394 |         pstmt.execute();

```

```

389 | private void btnSimpanActionPerformed(java.awt.event.ActionEvent evt) {
390 |     try {
391 |         String sql = "INSERT INTO tb_dokumen VALUES ('"+txtkodedokumen.getText()+"','"+txtnamadokumen.getText()+"','"+cmbkatagoridokumen.getSelectedItem()+"','"+txtlokasi.get
392 |         java.sql.Connection conn = (Connection)Konfig.configDB();
393 |         java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
394 |         pstmt.execute();
395 |         JOptionPane.showMessageDialog(null, "Proses simpan Data berhasil");
396 |         tampilkan_data();
397 |         kosongkan_from();
398 |     } catch (HeadlessException | SQLException e) {
399 |         JOptionPane.showMessageDialog(this, e.getMessage());
400 |     }
401 | }
402 |
403 |
404 | private void tabledokumenMouseClicked(java.awt.event.MouseEvent evt) {
405 |     int baris = tabledokumen.rowAtPoint(evt.getPoint());
406 |     String nik = tabledokumen.getValueAt(baris, 1).toString();
407 |     txtkodedokumen.setText(nik);
408 |
409 |     String nama = tabledokumen.getValueAt(baris, 2).toString();
410 |     txtnamadokumen.setText(nama);
411 |
412 |     String jab = tabledokumen.getValueAt(baris, 3).toString();
413 |     cmbkatagoridokumen.setSelectedItem(jab);

```

```

409 |     String nama = tabledokumen.getValueAt(baris, 2).toString();
410 |     txtnamadokumen.setText(nama);
411 |
412 |     String jab = tabledokumen.getValueAt(baris, 3).toString();
413 |     cmbkatagoridokumen.setSelectedItem(jab);
414 |
415 |     String alamat = tabledokumen.getValueAt(baris, 4).toString();
416 |     txtlokasi.setText(alamat);
417 |
418 |     String email = tabledokumen.getValueAt(baris, 5).toString();
419 |     txtdeskripsi.setText(email);
420 |
421 |     String telp = tabledokumen.getValueAt(baris, 6).toString();
422 |     txttanggal.setText(telp);
423 | }
424 |
425 | private void btnEditActionPerformed(java.awt.event.ActionEvent evt) {
426 |     try {
427 |         String sql = "UPDATE tb_dokumen SET kode = '"+txtkodedokumen.getText()+"',nama = '"+txtnamadokumen.getText()+"',kategori = '"+cmbkatagoridokumen.getSelectedItem()+"',
428 |         java.sql.Connection conn = (Connection)Konfig.configDB();
429 |         java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
430 |         pstmt.execute();
431 |         JOptionPane.showMessageDialog(null, "Edit Data Berhasil");
432 |     } catch (HeadlessException | SQLException e) {
433 |         JOptionPane.showMessageDialog(this, e.getMessage());

```

```

432     } catch (HeadlessException | SQLException e) {
433         JOptionPane.showMessageDialog(this, e.getMessage());
434     }
435     tampilkan_data();
436     kosongkan_from();
437 }
438
439 private void btnHapusActionPerformed(java.awt.event.ActionEvent evt) {
440     try {
441         String sql = "DELETE FROM tb_dokumen WHERE kode = '"+txtkodedokumen.getText()+"'";
442         java.sql.Connection conn = (Connection)Konfig.configDB();
443         java.sql.PreparedStatement pstmt = conn.prepareStatement(sql);
444         pstmt.execute();
445         JOptionPane.showMessageDialog(null, "Hapus Data Berhasil");
446     } catch (HeadlessException | SQLException e) {
447         JOptionPane.showMessageDialog(this, e.getMessage());
448     }
449     tampilkan_data();
450     kosongkan_from();
451 }
452
453 private void txtcariActionPerformed(java.awt.event.ActionEvent evt) {
454     // TODO add your handling code here:
455 }
456
457 private void txtcariKeyPressed(java.awt.event.KeyEvent evt) {

```

```

451     }
452
453 private void txtcariActionPerformed(java.awt.event.ActionEvent evt) {
454     // TODO add your handling code here:
455 }
456
457 private void txtcariKeyPressed(java.awt.event.KeyEvent evt) {
458     tampilkan_data();
459 }
460
461 private void txtnamadokumenActionPerformed(java.awt.event.ActionEvent evt) {
462     // TODO add your handling code here:
463 }
464
465 private void btnKeluarActionPerformed(java.awt.event.ActionEvent evt) {
466     // TODO add your handling code here:
467     int pilihan = JOptionPane.showConfirmDialog(this, "Apakah anda yakin?", "Konfirmasi", JOptionPane.YES_NO_OPTION);
468     if (pilihan == 0) {
469         Login login = new Login();
470         login.setExtendedState(Frame.MAXIMIZED_BOTH);
471         login.setVisible(true);
472         this.setVisible(false);
473     }
474 }
475

```

Hasil akhirnya adalah sbb

The image shows a Windows 10 desktop environment. The top window is a 'Login' form with a blue background. It contains fields for 'Username' (with 'admin' entered) and 'Password' (with '\*\*\*\*\*' entered), and a 'Masuk' button. The bottom window is titled 'Form Dokumen' and contains a search bar 'Cari Nama', several input fields for document details, a table with 4 columns, and buttons for data management.

**Login Form**

Username  
admin

Password  
\*\*\*\*\*

Masuk

**Form Dokumen**

Cari Nama

Kode Dokumen

Nama Dokumen

Kategori Dokumen  
Umum

Lokasi Dokumen

Deskripsi Dokumen

Tanggal

Tambah Data Simpan Edit hapus

Title 1	Title 2	Title 3	Title 4

keluar