

LAPORAN PRAKTIK PERTEMUAN KETIGA BELAS

Disusun guna memenuhi tugas mata kuliah Pemrograman Berorientasi Obyek
yang di ampu oleh:

Bayu Adhi Nugroho, Ph. D.



**UIN SUNAN AMPEL
S U R A B A Y A**

Disusun oleh:

Fernando Seroy (09040624087)

PROGRAM STUDI SISTEM INFORMASI

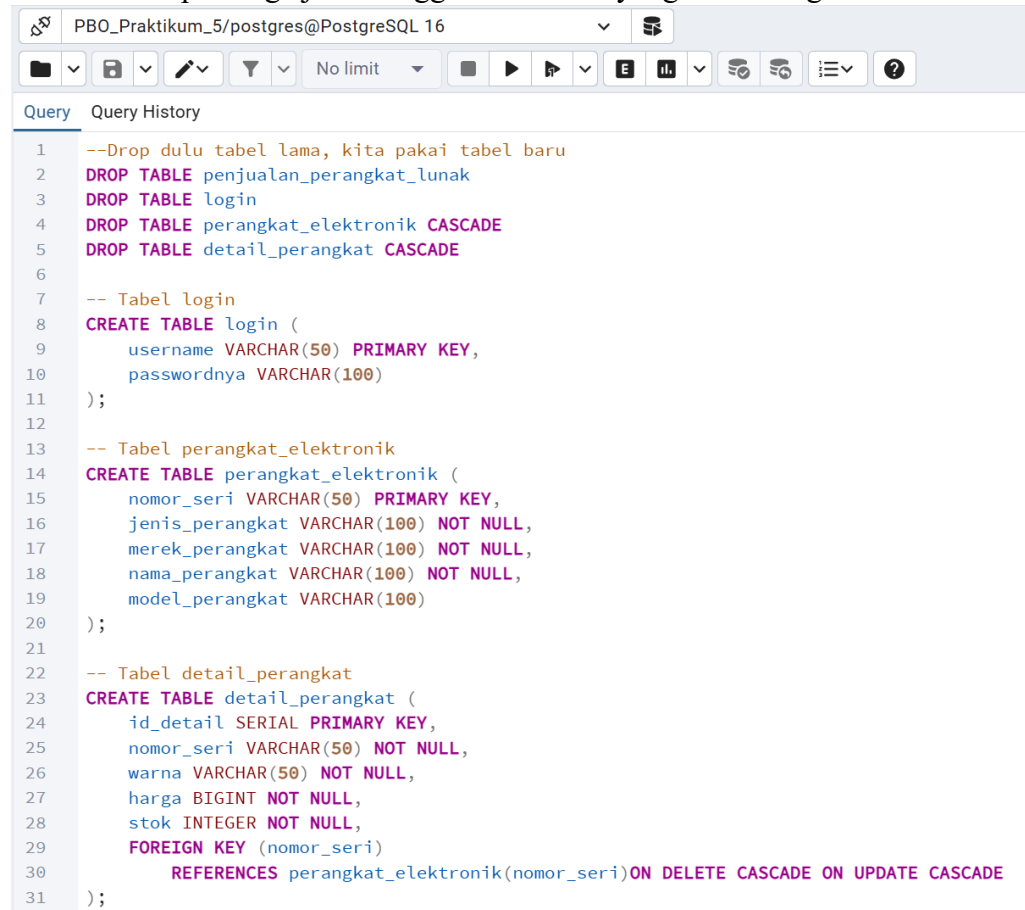
FAKULTAS SAINS DAN TEKNOLOGI

UNIVERSITAS ISLAM NEGERI SUNAN AMPEL SURABAYA

2025

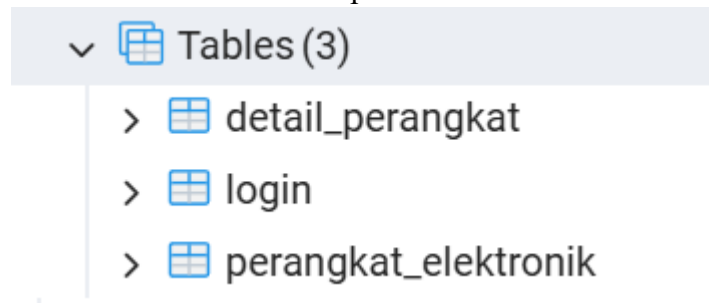
1. Perubahan Database

- 1.1. Kita tetap menggunakan database dengan koneksi yang sama di PostgreSQL, namun yang berubah adalah tabelnya, aku berikan sedikit bantuan atau privilege jika menggunakan tabel yang sama denganku:














```
1  --Drop dulu tabel lama, kita pakai tabel baru
2  DROP TABLE penjualan_perangkat_lunak
3  DROP TABLE login
4  DROP TABLE perangkat_elektronik CASCADE
5  DROP TABLE detail_perangkat CASCADE
6
7  -- Tabel login
8  CREATE TABLE login (
9      username VARCHAR(50) PRIMARY KEY,
10     passwordnya VARCHAR(100)
11 );
12
13 -- Tabel perangkat_elektronik
14 CREATE TABLE perangkat_elektronik (
15     nomor_seri VARCHAR(50) PRIMARY KEY,
16     jenis_perangkat VARCHAR(100) NOT NULL,
17     merek_perangkat VARCHAR(100) NOT NULL,
18     nama_perangkat VARCHAR(100) NOT NULL,
19     model_perangkat VARCHAR(100)
20 );
21
22 -- Tabel detail_perangkat
23 CREATE TABLE detail_perangkat (
24     id_detail SERIAL PRIMARY KEY,
25     nomor_seri VARCHAR(50) NOT NULL,
26     warna VARCHAR(50) NOT NULL,
27     harga BIGINT NOT NULL,
28     stok INTEGER NOT NULL,
29     FOREIGN KEY (nomor_seri)
30         REFERENCES perangkat_elektronik(nomor_seri) ON DELETE CASCADE ON UPDATE CASCADE
31 );
```

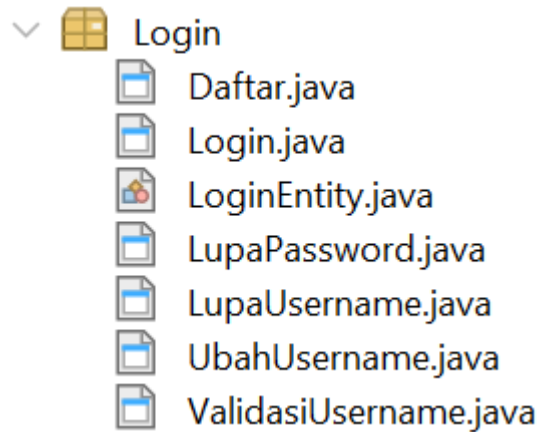
- 1.2. Maka setelah kamu menjalankan query sql tersebut, refresh table nya nanti akan muncul hasil seperti ini:



- 1.3. Jangan lupa isi tabel perangkat_elektroniknya terlebih dahulu, karena tabel tersebut adalah tabel utama. Lalu bisa kita isi tabel detail_perangkat via aplikasi yang kita buat saat ini. Karena fokus kita sudah berganti pada fitur "Login", maka kita perlu membuat package baru dan class baru. Untuk cara pengisian tabel perangkat_elektronik

Data Output			Messages	Notifications					
									SQL
	username [PK] character varying (50) 	passwordnya character varying (100) 							
1	verroy01	passA7x!							
2	user_alpha	alpha12345							
3	rinzoku	Rk#2025!							
4	techmaster	tm_secure88							
5	nova_user	Nv!2024pass							
6	skyline	skyPwd77							
7	aurora_dev	Aur0ra!dev							
8	bytehunter	Byte#512							
9	moonlight	MI!shadow22							
10	zenith_admin	ZAdmin#909							
11	Admin	ackviewqnv							
12	BROOOO	BENARCOY							

2.1. Disini, saya menggunakan package baru lalu memiliki 6 Class Jfroom dan 1 Class Entity:



2.1.1. Daftar.java (Jfrom)

```

1  package Login;
2  /**
3   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
4   * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template
5   */
6
7  import javax.persistence.*;
8  import javax.swing.JOptionPane;
9
10 /**
11  *
12  * @author LEGION
13  */
14 public class Daftar extends javax.swing.JFrame {
15
16     private static final java.util.logging.Logger logger = java.util.logging.Logger.getLogger(Daftar.class.getName());
17     private EntityManagerFactory emf;
18     private EntityManager em;
19
20     public Daftar() {
21         // ...5 lines ...
22     }
23
24     private void testDatabaseConnection() {
25         // ...10 lines ...
26     }
27
28     private void connectToDatabase() {
29         // ...8 lines ...
30     }
31
32     private void centerFrame() {
33         // ...3 lines ...
34     }
35
36     private boolean isUsernameExists(String username) {
37         // ...10 lines ...
38     }
39
40     private boolean registerUser(String username, String password) {
41         // ...22 lines ...
42     }
43
44     @Override
45     public void dispose() {
46         // ...13 lines ...
47     }
48
49     /**
50      * This method is called from within the constructor to initialize the form.
51      * WARNING: Do NOT modify this code. The content of this method is always
52      * regenerated by the Form Editor.
53      */
54     @SuppressWarnings("unchecked")
55     Generated Code

```

2.1.2. Login.java (Jfrom)

```

1 1
2 2
3 3
4 4
5 package Login;
6
7 import javax.persistence.*;
8 import javax.swing.JOptionPane;
9
10 /**
11  *
12  * @author LEGION
13  */
14 public class Login extends javax.swing.JFrame {
15
16     private static final java.util.logging.Logger logger = java.util.logging.Logger.getLogger(Login.class.getName());
17     private EntityManagerFactory emf;
18     private EntityManager em;
19
20     /**
21      * Creates new form Login
22      */
23     public Login() {
24         // ...5 lines
25     }
26
27     private void testDatabaseConnection() {
28         // ...10 lines
29     }
30
31     private void connectToDatabase() {
32         // ...10 lines
33     }
34
35     private void centerFrame() {
36         // ...3 lines
37     }
38
39     private boolean validateLogin(String username, String password) {
40         // ...13 lines
41     }
42
43     private void openMainApplication() {
44         // ...5 lines
45     }
46
47     /**
48      * This method is called from within the constructor to initialize the form.
49      * WARNING: Do NOT modify this code. The content of this method is always
50      * regenerated by the Form Editor.
51      */
52 }

```

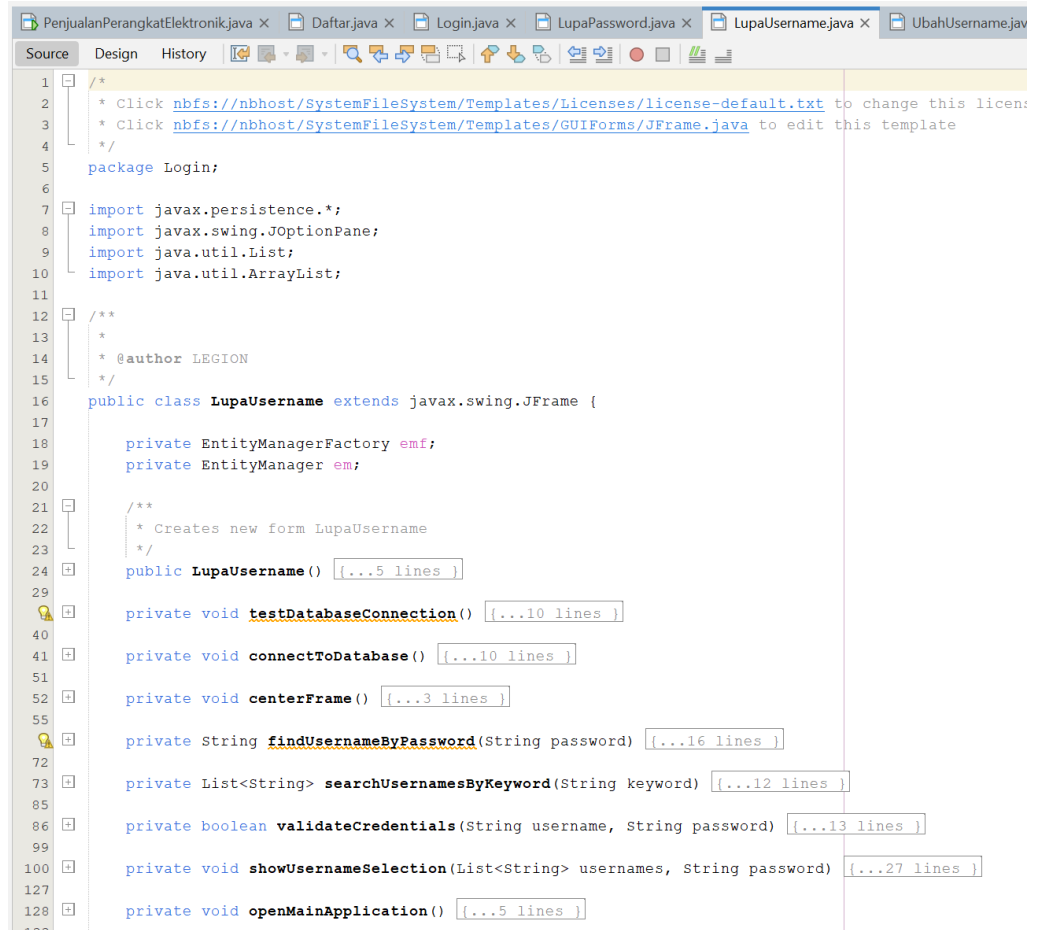
2.1.3. LupaPassword.java (Jfrom)

```

1 1
2 2
3 3
4 4
5 package Login;
6
7 import javax.persistence.*;
8 import javax.swing.JOptionPane;
9 import javax.swing.SwingUtilities;
10
11 /**
12  *
13  * @author LEGION
14  */
15 public class LupaPassword extends javax.swing.JFrame {
16
17     private EntityManagerFactory emf;
18     private EntityManager em;
19     private String username;
20
21     public LupaPassword(String username) {
22         // ...19 lines
23     }
24
25     private void connectToDatabase() {
26         // ...8 lines
27     }
28
29     private void centerFrame() {
30         // ...3 lines
31     }
32
33     private boolean isUsernameExists(String username) {
34         // ...12 lines
35     }
36
37     private boolean updatePassword(String username, String newPassword) {
38         // ...36 lines
39     }
40
41     private void addUsernameValidation() {
42         // ...30 lines
43     }
44
45     @Override
46     public void dispose() {
47         // ...13 lines
48     }
49 }

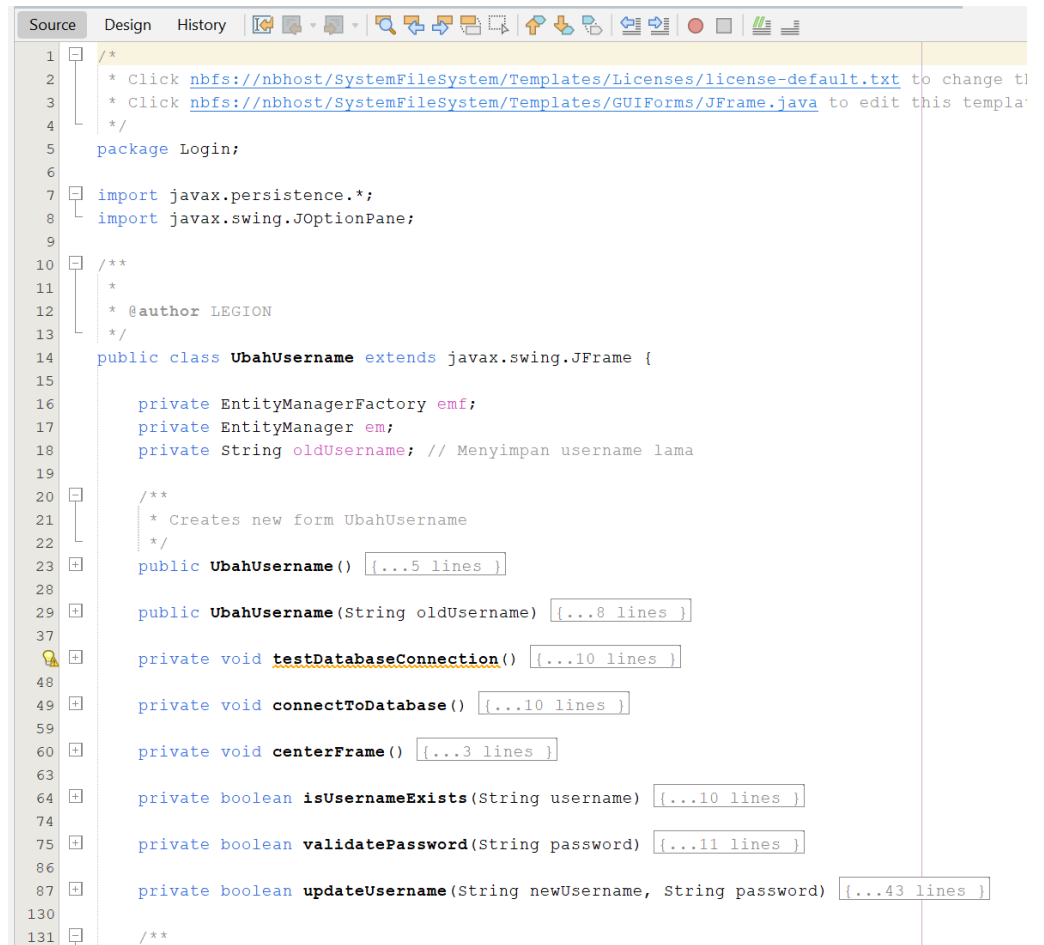
```

2.1.4. LupaUsername.java (Jfrom)



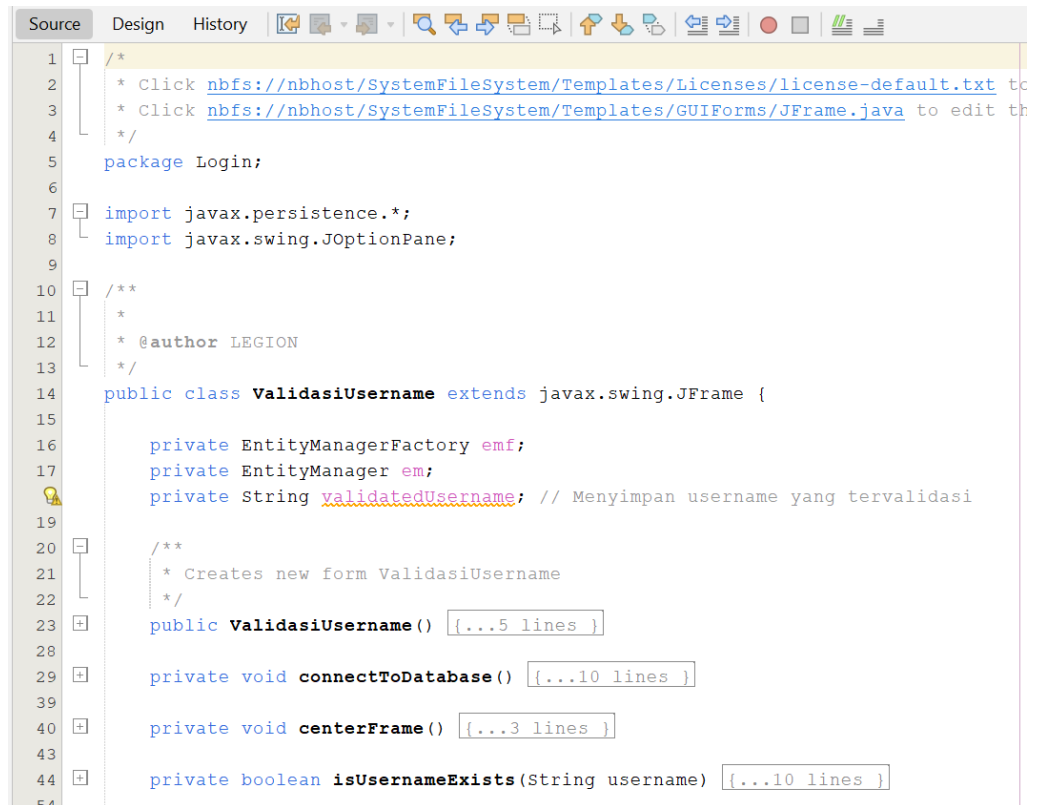
```
1  /*
2  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3  * Click nbfs://nbhost/SystemFileSystem/Templates/GuiForms/JFrame.java to edit this template
4  */
5  package Login;
6
7  import javax.persistence.*;
8  import javax.swing.JOptionPane;
9  import java.util.List;
10 import java.util.ArrayList;
11
12 /**
13 *
14 * @author LEGION
15 */
16 public class LupaUsername extends javax.swing.JFrame {
17
18     private EntityManagerFactory emf;
19     private EntityManager em;
20
21     /**
22      * Creates new form LupaUsername
23      */
24     public LupaUsername() {
25         // ...5 lines ...
26     }
27
28     private void testDatabaseConnection() {
29         // ...10 lines ...
30     }
31
32     private void connectToDatabase() {
33         // ...10 lines ...
34     }
35
36     private void centerFrame() {
37         // ...3 lines ...
38     }
39
40     private String findUsernameByPassword(String password) {
41         // ...16 lines ...
42     }
43
44     private List<String> searchUsernamesByKeyword(String keyword) {
45         // ...12 lines ...
46     }
47
48     private boolean validateCredentials(String username, String password) {
49         // ...13 lines ...
50     }
51
52     private void showUsernameSelection(List<String> usernames, String password) {
53         // ...27 lines ...
54     }
55
56     private void openMainApplication() {
57         // ...5 lines ...
58     }
59 }
```

2.1.5. UbahUsername.java (Jfrom)



```
1  /**
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change the license
3   * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template
4   */
5  package Login;
6
7  import javax.persistence.*;
8  import javax.swing.JOptionPane;
9
10 /**
11  *
12  * @author LEGION
13  */
14 public class UbahUsername extends javax.swing.JFrame {
15
16     private EntityManagerFactory emf;
17     private EntityManager em;
18     private String oldUsername; // Menyimpan username lama
19
20     /**
21     * Creates new form UbahUsername
22     */
23     public UbahUsername() { ...5 lines }
24
25     public UbahUsername(String oldUsername) { ...8 lines }
26
27     private void testDatabaseConnection() { ...10 lines }
28
29     private void connectToDatabase() { ...10 lines }
30
31     private void centerFrame() { ...3 lines }
32
33     private boolean isUsernameExists(String username) { ...10 lines }
34
35     private boolean validatePassword(String password) { ...11 lines }
36
37     private boolean updateUsername(String newUsername, String password) { ...43 lines }
38
39     /**
40     */
41 }
```

2.1.6. ValidasiUsername.java (Jfrom)



```
1  /**
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to
3   * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit th
4   */
5   package Login;
6
7   import javax.persistence.*;
8   import javax.swing.JOptionPane;
9
10  /**
11   *
12   * @author LEGION
13   */
14  public class ValidasiUsername extends javax.swing.JFrame {
15
16      private EntityManagerFactory emf;
17      private EntityManager em;
18      private String validatedUsername; // Menyimpan username yang tervalidasi
19
20      /**
21       * Creates new form ValidasiUsername
22       */
23      public ValidasiUsername() { ...5 lines }
24
25      private void connectToDatabase() { ...10 lines }
26
27      private void centerFrame() { ...3 lines }
28
29      private boolean isUsernameExists(String username) { ...10 lines }
30
31  }
```

2.1.7. LoginEntity.java (Entity Class)


```

Source History
1  /**
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3   * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit this template
4   */
5   package Login;
6
7   import java.io.Serializable;
8   import javax.persistence.Basic;
9   import javax.persistence.Column;
10  import javax.persistence.Entity;
11  import javax.persistence.Id;
12  import javax.persistence.NamedQueries;
13  import javax.persistence.NamedQuery;
14  import javax.persistence.Table;
15
16  /**
17   *
18   * @author LEGION
19   */
20  @Entity
21  @Table(name = "login")
22  @NamedQueries({
23      @NamedQuery(name = "LoginEntity.findAll", query = "SELECT l FROM LoginEntity l"),
24      @NamedQuery(name = "LoginEntity.findByUsername", query = "SELECT l FROM LoginEntity l WHERE l.username = ?1"),
25      @NamedQuery(name = "LoginEntity.findByUsernameAndPassword", query = "SELECT l FROM LoginEntity l WHERE l.username = ?1 AND l.passwordnya = ?2")
26  })
27  public class LoginEntity implements Serializable {
28
29      private static final long serialVersionUID = 1L;
30
31      @Id
32      @Column(name = "username", length = 50)
33      private String username;
34
35      @Column(name = "passwordnya", length = 100)
36      private String passwordnya;
37
38      public LoginEntity() {
39      }
40
41      public LoginEntity(String username, String passwordnya) {
42          this.username = username;

```

2.2. Jangan lupa untuk mengubah code di main class

PenjualanPerangkatElektronik.java dengan source code ini agar disaat pertama kali me run file, kita akan diarahkan kedalam class login terlebih dahulu.

```

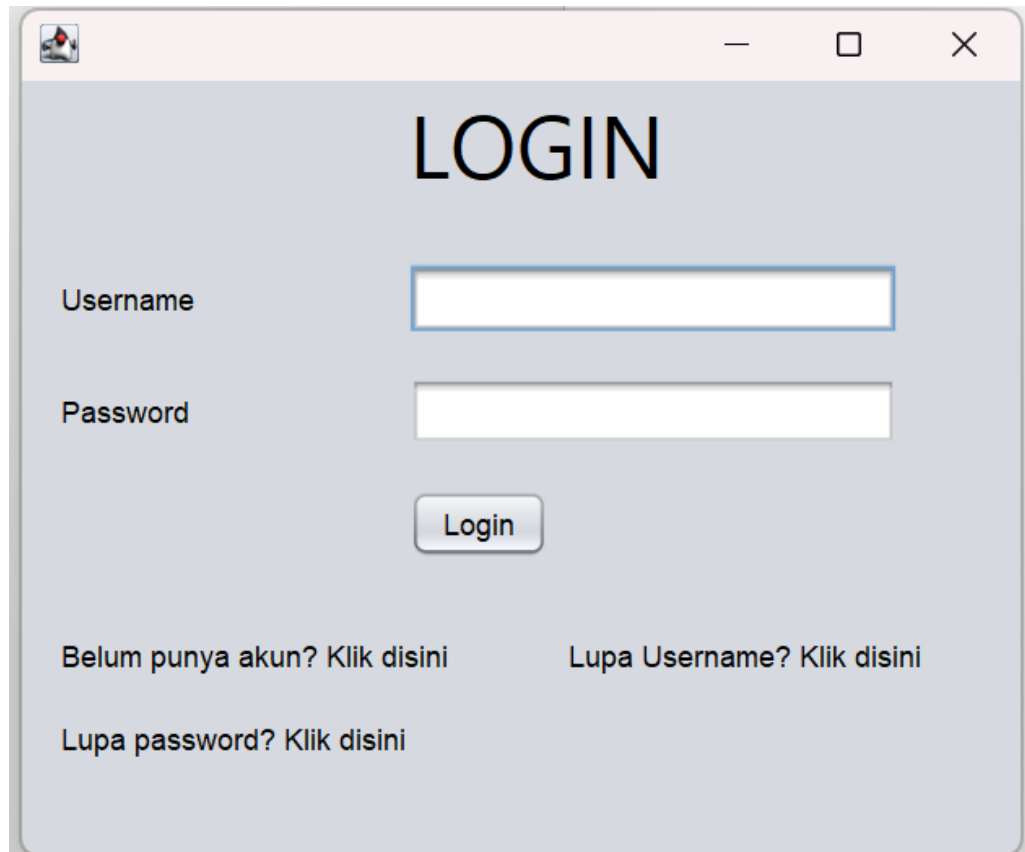
public static void main(String args[]) {
    /* Set the Nimbus look and feel */
    //<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">
    /* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.
     * For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
     */
    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 | InstantiationException | IllegalAccessException | javax.swing.UnsupportedLookAndFeelException ex) {
        logger.log(java.util.logging.Level.SEVERE, null, ex);
    }
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(() -> {
        Login.Login loginForm = new Login.Login();
        loginForm.setVisible(true);
    });
}

```

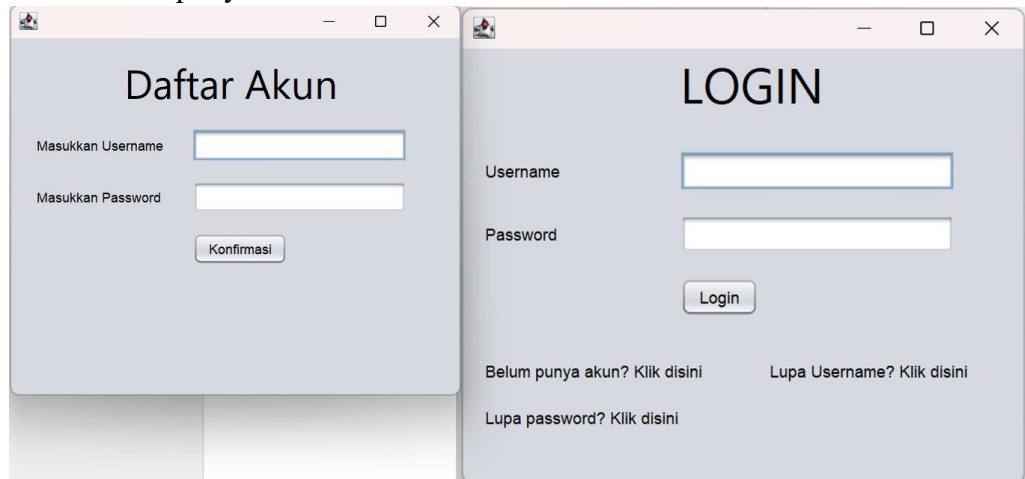
3. Output

3.1. Kita run di main Class terlebih dahulu, dan ternyata alur yang kita pakai berhasil membuka menu login terlebih dahulu.



A screenshot of a web application window titled "LOGIN". The window has a light blue header bar with standard window controls (minimize, maximize, close). The main content area is light gray. At the top, the word "LOGIN" is displayed in large, bold, black capital letters. Below it, there are two input fields: "Username" and "Password", each with a white text box and a blue border. A "Login" button is positioned below the password field. At the bottom, there are three links: "Belum punya akun? Klik disini", "Lupa Username? Klik disini", and "Lupa password? Klik disini".

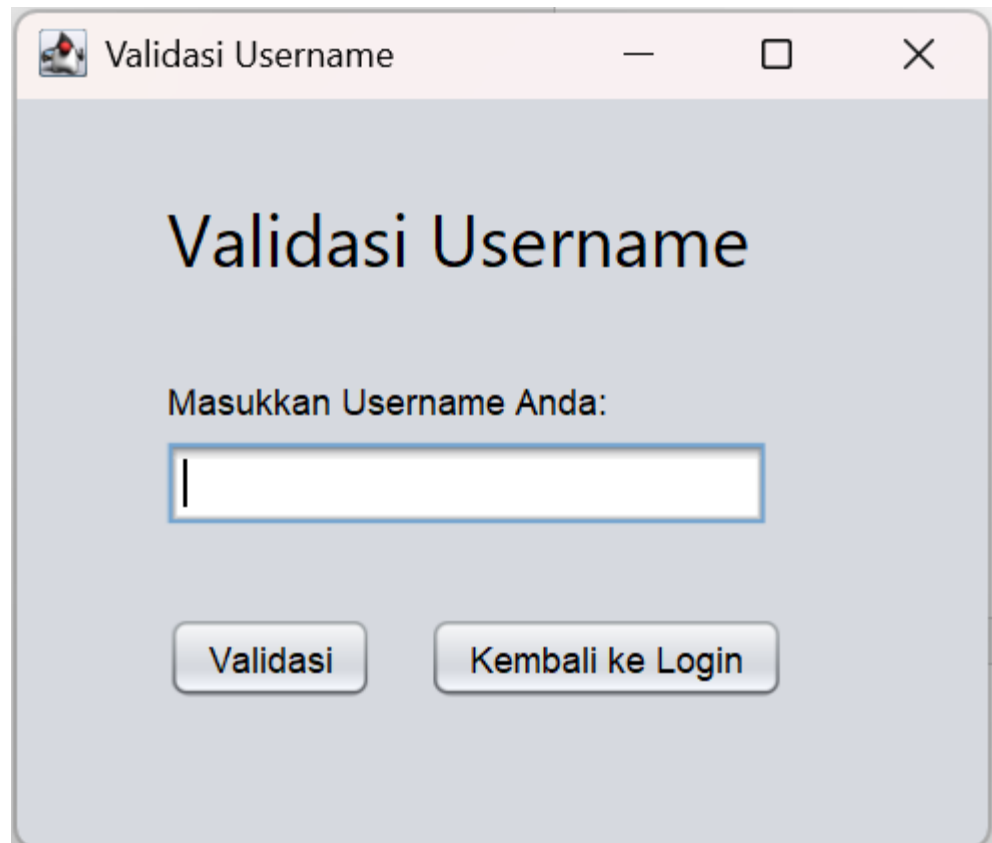
3.2. Cek “Belum punya akun? Klik disini”



A screenshot showing two overlapping web application windows. The window in the foreground is titled "Daftar Akun" (Register Account). It has a light blue header bar and a light gray main content area. The title "Daftar Akun" is in large, bold, black capital letters. Below it, there are two input fields: "Masukkan Username" and "Masukkan Password", each with a white text box and a blue border. A "Konfirmasi" (Confirm) button is positioned below the password field. The window in the background is the "LOGIN" window, which is partially obscured by the "Daftar Akun" window. It shows the same "LOGIN" title, input fields, and links as the first screenshot.

Ternyata work.

3.3. Cek “Lupa password? Klik disini”



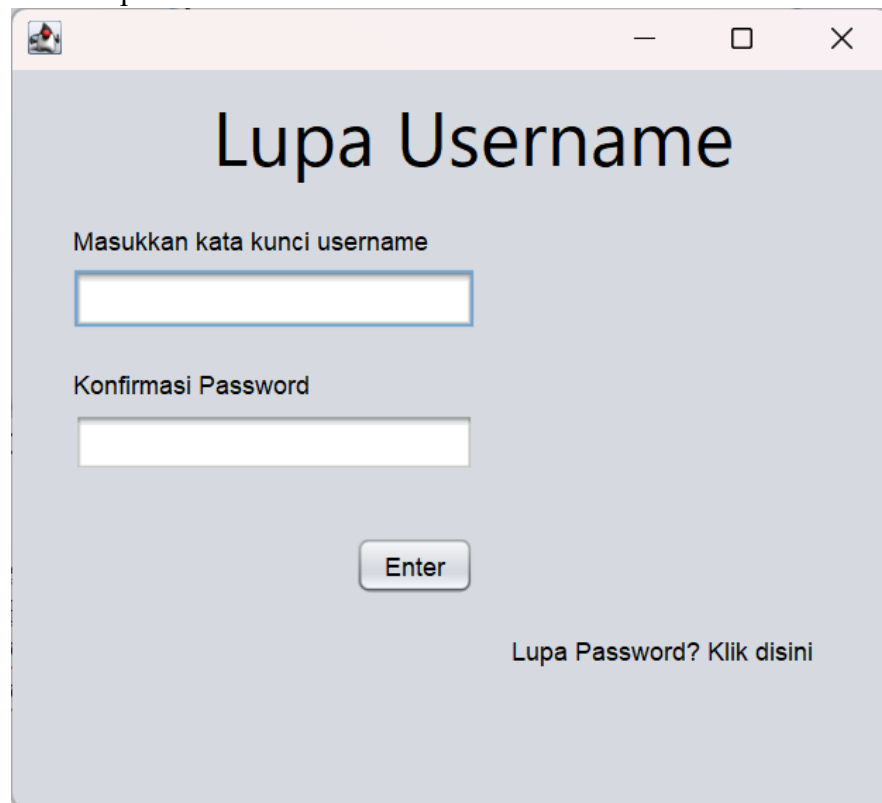
Validasi Username

Validasi Username

Masukkan Username Anda:

Validasi Kembali ke Login

3.4. Cek “Lupa Username? Klik disini”



Lupa Username

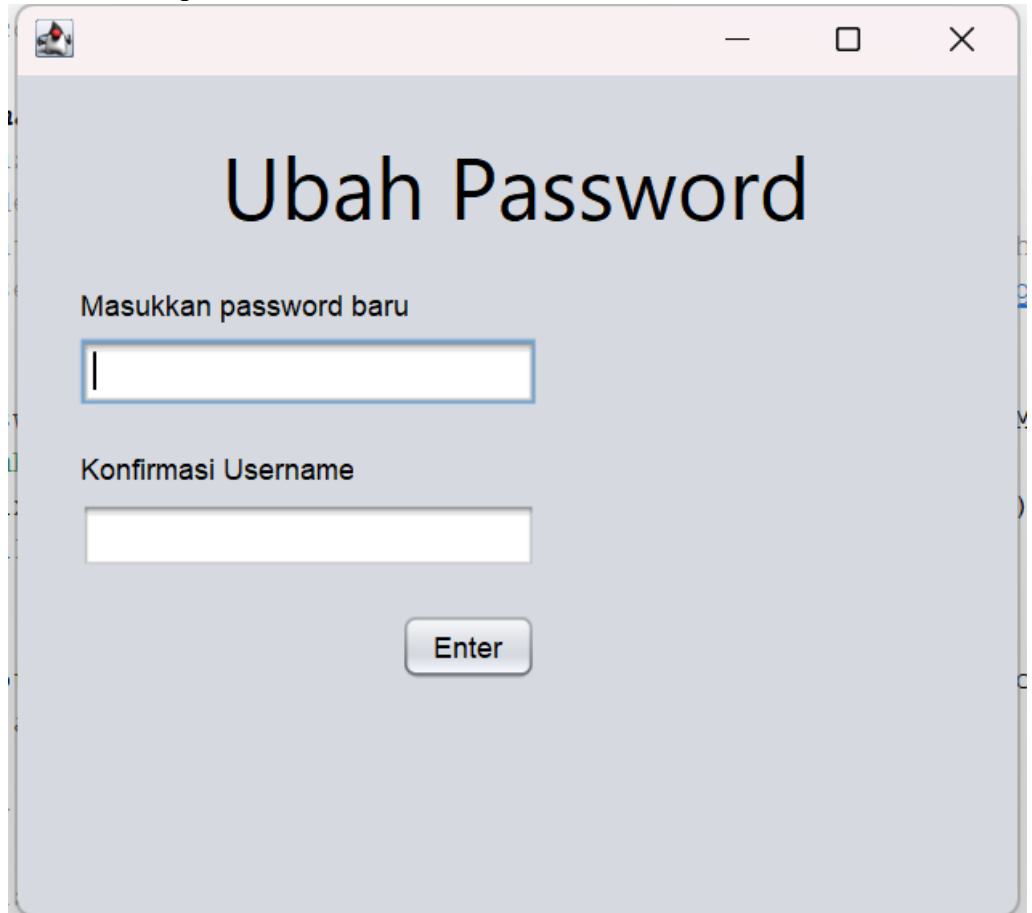
Masukkan kata kunci username

Konfirmasi Password

Enter

Lupa Password? Klik disini

- 3.4.1. Disini alurnya adalah kita harus mengetahui passwordnya karena dari password bisa terdeteksi usernamenya, kemungkinan username sama namun bisa saja password berbeda. Namun tidak menutup kemungkinan untuk duplikasi username dan password secara tidak sengaja maupun disengaja. Karena belum ditambahkan fitur username tidak boleh sama.
Cek fitur "Lupa Password? Klik disini"



The image shows a screenshot of a web application window titled "Ubah Password". The window has a light gray background and a standard Windows-style title bar with a minimize button, a maximize button, and a close button. The main content area contains the following elements:

- A large heading "Ubah Password" in a bold, black, sans-serif font.
- A label "Masukkan password baru" (Enter new password) in a smaller, black, sans-serif font.
- A text input field with a blue border and a vertical cursor, positioned below the label.
- A label "Konfirmasi Username" (Confirm Username) in a smaller, black, sans-serif font.
- A text input field with a gray border, positioned below the label.
- An "Enter" button with a light gray background and a dark gray border, positioned below the second input field.

- 3.5. Tampilan bila berhasil login

DATA PENJUALAN PERANGKAT

Tabel Perangkat
Detail Perangkat

Nomor Seri	Jenis Perangkat	Merek Perangkat	Nama Perangkat	Model Perangkat
L001	Laptop	Lenovo	IdeaPad Slim 5i	82LN004AID
L002	Laptop	HP	Pavilion Aero 13	13-be2047AU
L003	Laptop	ASUS	Vivobook S14	K3402ZA-OLED552
L004	Laptop	Acer	Swift X 14	SFX14-71G-78V3
L005	Laptop	Dell	Inspiron 14 Plus	INP143012SL
L006	Laptop	MSI	Modern 15 A13M	A13M-242ID

Insert
Update
Upload

Delete
Search
Cetak
Refresh

3.6. Coba Cetak file jasper

LAPORAN GABUNGAN PERANGKAT ELEKTRONIK
 Data Master dan Detail Perangkat

A. DATA MASTER PERANGKAT ELEKTRONIK

Nomor Seri	Jenis	Merek	Nama	Model
L001	Laptop	Lenovo	IdeaPad Slim 5i	82LN004AID
Warna: Hitam	Harga: Rp 8.999.000	Stok: 7	Status: Tersedia	
L002	Laptop	HP	Pavilion Aero 13	13-be2047AU
Warna: Silver	Harga: Rp 9.999.000	Stok: 5	Status: Tersedia	
L003	Laptop	ASUS	Vivobook S14	K3402ZA-OLED552
Warna: Abu-abu	Harga: Rp 10.500.000	Stok: 6	Status: Tersedia	
L004	Laptop	Acer	Swift X 14	SFX14-71G-78V3
Warna: Biru Gelap	Harga: Rp 11.250.000	Stok: 4	Status: Tersedia	
L005	Laptop	Dell	Inspiron 14 Plus	INP143012SL
Warna: Putih	Harga: Rp 9.500.000	Stok: 11	Status: Tersedia	
L006	Laptop	MSI	Modern 15 A13M	A13M-242ID
Warna: Merah	Harga: Rp 9.800.000	Stok: 3	Status: Tersedia	

Semua Source code berjalan dengan lancar.