

LAPORAN PRAKTIKUM
Pemograman Berbasis Objek

JUDUL : PRAKTIKUM 1



Disusun oleh:
Ndaru Pradana Gatot Suseno - 2411532007

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
UNIVERSITAS ANDALAS
SEPTEMBER
2025

A. Pendahuluan

1. Tujuan

Tujuan praktikum ini yaitu mahasiswa mampu membuat class, object, encapsulation, constructor

dan method menggunakan Bahasa java dan menerapkannya pada aplikasi manajemen laundry,

Adapun pada praktikum ini mahasiswa akan mempelajari beberapa poin yaitu :

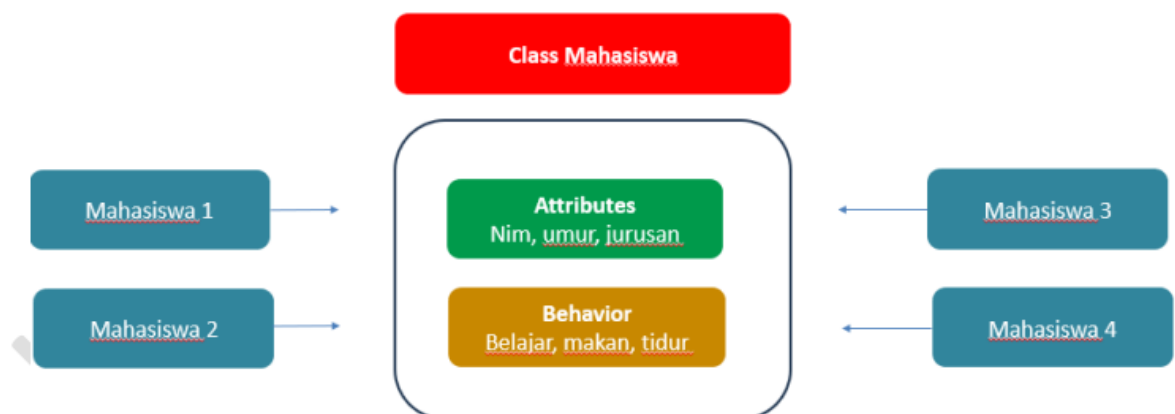
- Membuat class, object, encapsulation, constructor dan method untuk aplikasi laundry seperti class User, Costumer, Service dan Order
- Membuat desain antarmuka aplikasi Laundry Login dan Halaman Utama
- Mampu menggunakan method pada JFrame/Tampilan aplikasi
- Mampu membuat fungsi berpindah dari tampilan login ke halaman utama

2. Alat

- Computer / laptop yang telah terinstall JDK dan Eclipse

3. Teori

Class merupakan sekumpulan objek yang memiliki karakteristik/sifat/behavior dan properties/atribut yang sama, class juga bisa disebut sebagai template/blueprint dari objek yang akan dibuat. Misalkan misalkan ada class mahasiswa, yaitu nama-nama mahasiswa misalkan fulan merupakan sebuah object.



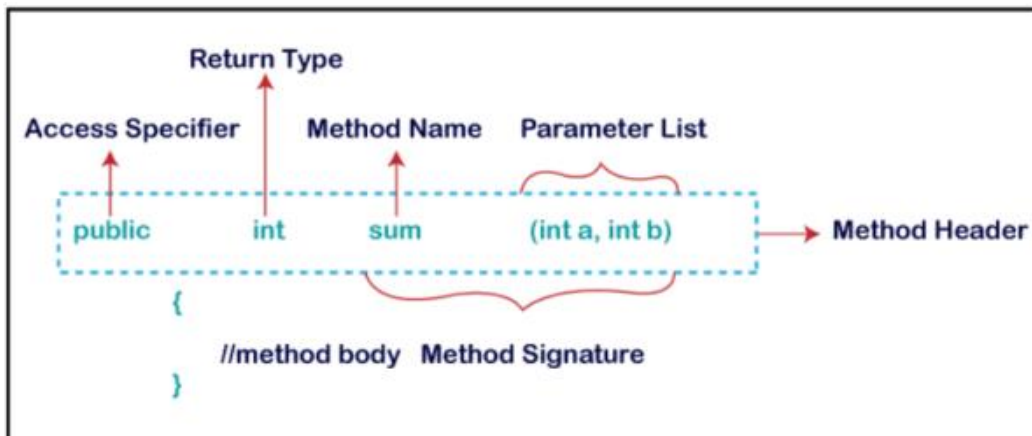
Sebuah class bukan sebuah entitas didalam dunia nyata melainkan sebagai blueprint untuk membuat suatu objek, class didalam java dapat berisi Data Member, Method, Constructor, Nested Class dan Interface.

Object merupakan representasi entitis yang ada pada dunia nyata, sebuah object terdiri dari :

- State merepresentasikan attribute dari sebuah object
- Behavior merepresentasikan method dari sebuah object
- Identity nama yang unik sebuah objek dan digunakan untuk interaksi dengan objek lain.



Method merupakan sebuah blok kode yang dapat dijalankan berulang kali, sehingga lebih terorganisir dan dapat digunakan Kembali, contoh method yang disediakan oleh java adalah method main(), equals(), toString() dll. Pola sebuah method dapat dilihat seperti gambar berikut :



Aturan pembuatan method pada java :

- Penamaan method harus menggunakan **verb** dan dimulai dengan **lowercase** letter
- Jika nama method lebih dari satu kata maka kata pertama harus **verb** dan selanjutnya **adjective** atau **noun** seperti sum(), area()
- jika nama method terdiri lebih dari 1 kata, maka kata pertama **lowercase** dan kata selanjutnya **uppercase** seperti areaOfCircle()

Method pada java terdiri dari beberapa jenis, berikut jenis-jenis method yang ada pada Java.

- Predefined Method (standard library method or built-in method) method yang sudah disediakan oleh java seperti **length()**, **equals()**, **compareTo()**, **print()**
- User-defined Method yaitu method yang dibuat oleh user atau Programmer, method dibuat sesuai dengan kebutuhan
- Static method yaitu Method yang menggunakan static sebelum nama method, keutamaan method static yaitu dapat memanggil/menggunakan method tanpa harus membuat object terlebih dahulu.

MODUL PRAKTIKUM PBO - JAVA NURFIAH

- Instance method merupakan method non static yaitu harus membuat object terlebih dahulu Ketika akan menggunakannya. Instance method terdiri dari 2 jenis yaitu accessor method dan mutator method.

o Accessor Method digunakan untuk membaca instance, menggunakan kata kunci **get** atau disebut **getter**.

```
public String getNim() {
    return nim;
}
```

o Mutator Method digunakan untuk membaca dan mengubah nilai, menggunakan kata kunci **set** atau disebut juga dengan **setter** dan akan mengembalikan value yang bersifat private.

```
public void setNim(String nim) {
    this.nim = nim;
}
```

- Abstract Method yaitu method yang tidak memiliki body, dideklarasikan didalam class abstract dengan menggunakan kata kunci abstract.
- Factory Method yaitu method yang digunakan untuk mengembalikan object ke class yang bersangkutan, semua method static disebut factory method contohnya

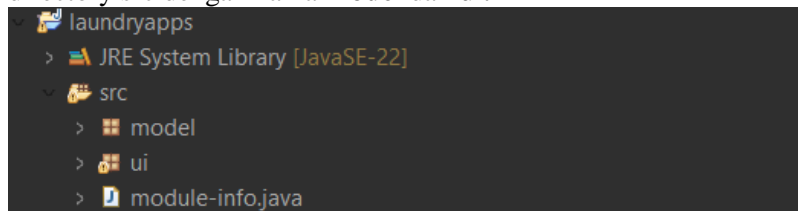
NumberFormat

obj = NumberFormat.getNumberInstance();

Constructor yaitu blok kode yang mirip dengan method, menggunakan kata kunci new Ketika akan membuat sebuah object, nama constructor harus sama dengan nama class.

B. Langkah-Langkah Pengerjaan

1. Membuat java Project dengan nama **Laundryapps**, kemudian buat 2 package pada directory src dengan nama **model** dan **ui**.



2. Membuat class baru di package **model** dengan nama **User**.



3. Membuat attribute class **User** berupa id, nama, username dan password.

```
1 package model;
2
3 public class User {
4     String id, nama, username, password;
5 }
```

4. Membuat **setter** dan **getter** attribute di class user, method ini berfungsi untuk memasukkan/mengubah dan menampilkan value dari suatu objek.

```
•public String getId() {
    return id;
}

•public void setId(String id) {
    this.id = id;
}

•public String getNama() {
    return nama;
}

•public void setNama(String nama) {
    this.nama = nama;
}

•public String getUsername() {
    return username;
}

•public void setUsername(String username) {
    this.username = username;
}

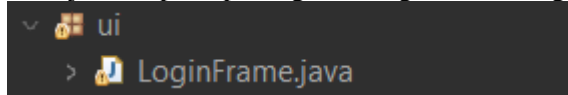
•public String getPassword() {
    return password;
}

•public void setPassword(String password) {
    this.password = password;
}
```

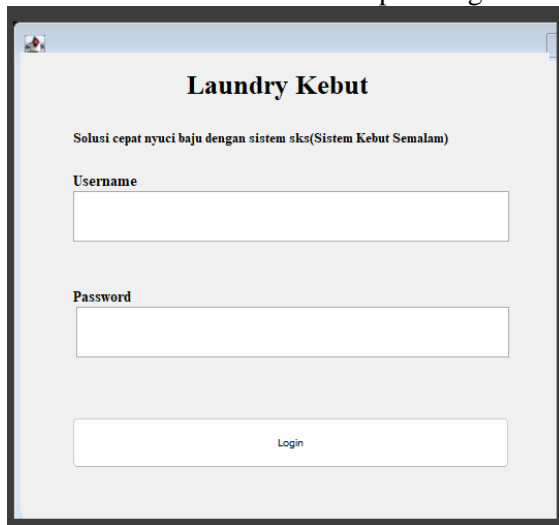
5. Membuat method login agar bisa digunakan oleh pengguna Ketika login ke aplikasi.

```
8•public static boolean login(String username, String password) {
9    boolean isLogin = false;
10   User user = new User();
11   user.setId("1");
12   user.setNama("fulan");
13   user.setUsername("fulan");
14   user.setPassword("12345");
15
16   if(user.getUsername().equalsIgnoreCase(username) && user.getPassword().equalsIgnoreCase(password)) {
17       isLogin = true;
18   } else {
19       isLogin = false;
20   }
21   return isLogin;
22 }
23
24
25 }
```

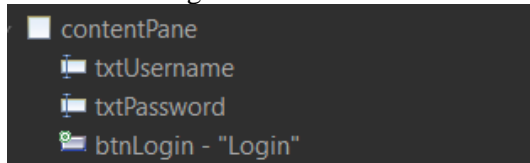
6. Buat JFrame pada package ui dengan nama LoginFrame.



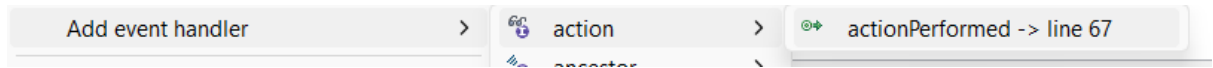
7. Kemudian membuat desain tampilan login.



8. Kemudian mengubah id dari attribut username, password dan JButton



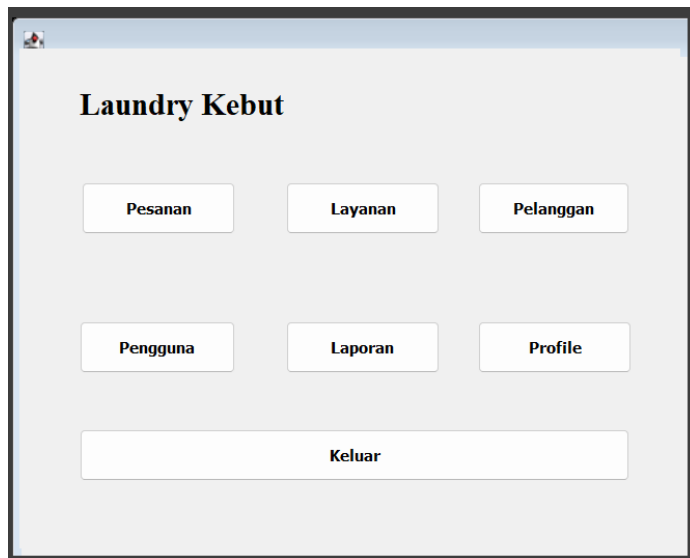
9. Langkah selanjutnya klik kanan pada button login, pilih add event handler > action > action Performed.



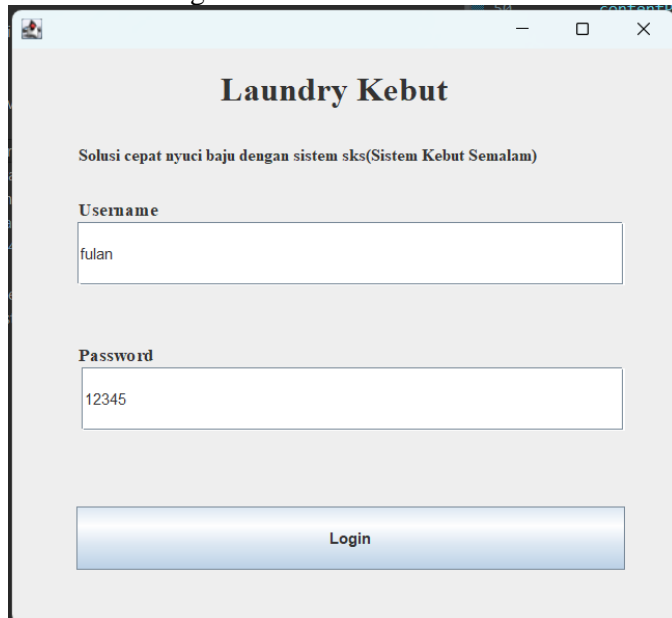
10. Kemudian panggil method login pada class User dengan mengirimkan parameter yang diambil dari txtUsername dan txtPassword, jika cocok maka akan tampil halaman utama.

```
 JButton btnLogin = new JButton("Login");
 btnLogin.addActionListener(new ActionListener() {
     public void actionPerformed(ActionEvent e) {
         if(User.login(txtUsername.getText(), txtPassword.getText())) {
             new MainFrame().setVisible(true);
             dispose();
         } else {
             JOptionPane.showMessageDialog(null, "Login Gagal");
         }
     }
 });
```

11. Buat JFrame baru pada package ui dengan nama MainFrame, lalu desain halaman utamanya dengan gambar dibawah ini.



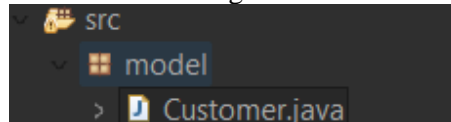
12. Jika semua Langkah diatas benar maka akan keluar hasil seperti ini.



C. Tugas

Buatlah class dengan nama Customer dengan attribute id, nama, Alamat dan nomor HP, buatlah setter dan getter untuk menambahkan object customer

1. Membuat class dengan nama Customer.



2. Tambahkan attribute id, nama, Alamat, dan nomor HP.

```
1 package model;
2
3 public class Customer {
4     String id,nama,alamat,nomorHP;
```

3. Buatlah setter dan getter untuk menambahkan object Customer.

```
5
6     public String getId() {
7         return id;
8     }
9
10    public void setId(String id) {
11        this.id = id;
12    }
13
14    public String getNama() {
15        return nama;
16    }
17
18    public void setNama(String nama) {
19        this.nama = nama;
20    }
21
22    public String getAlamat() {
23        return alamat;
24    }
25
26    public void setAlamat(String alamat) {
27        this.alamat = alamat;
28    }
29
30    public String getNomorHP() {
31        return nomorHP;
32    }
33
34    public void setNomorHP(String nomorHP) {
35        this.nomorHP = nomorHP;
36    }
37 }
38
```