Laporan Praktikum

Pemrograman Berorientasi Objek



Disusun Oleh :

**Ahda Rindang Al-Amin (2311531003)**

Dosen Pengampu : Nurfiah, S.ST, M.Kom.

Departemen Informatika

Fakultas Teknologi Informasi

Universitas Andalas

Tahun 2024

**Desain Aplikasi Laundry**

1. **Tujuan Praktikum**
2. Memahami kembali GUI menggunakan JFrame
3. Memahami penggunaan class, object, encapsulation, constructor, dan method pada Pemograman Berorientasi Objek
4. **Pendahuluan**

**Class**

Class merupakan sekumpulan objek yang memiliki karakteristik/sifat/beavior dan properties/atribut yang sama, class juga bisa disebut sebagai template/blueprint dari objek yang akan dibuat. Sebuah class bukan sebuah entitas di dalam dunia nyata melaikan sebagai blueprint untuk membuat suatu objek, class didalam java dapat berisi Data Member, Method, Constructor, Nested Class dan Interface.

**Object**

Object merupakan representasi entitis yang ada pada dunia nyata, sebuah object terdiri dari State sebagai atribut sebuah object, Behavior sebagai method, dan Identity.

**Method**

Method merupakan sebuah blok kode yang dapat dijalankan berulang kali, sehingga lebih terorganisir dan dapat digunakan kembali. Contoh method yang disediakan oleh library bawaan java adalah method main(), equals(), toString(), dll.

**Constructor**

Constructor adalah sebuah metode khusus yang digunakan untuk menginisialisasi objek dari sebuah kelas. Constructor biasanya dipanggil secara otomatis saat sebuah objek dibuat, dan bertugas untuk mengatur nilai awal dari atribut-atribut atau properti objek tersebut.

1. **Metode Praktikum**
2. Buat Java Project baru dengan nama laundryApps, dan buat 2 package untuk model dan ui.
3. Buat class baru pada model dengan nama User dan tambahkan atribut id, nama, username, dan password.

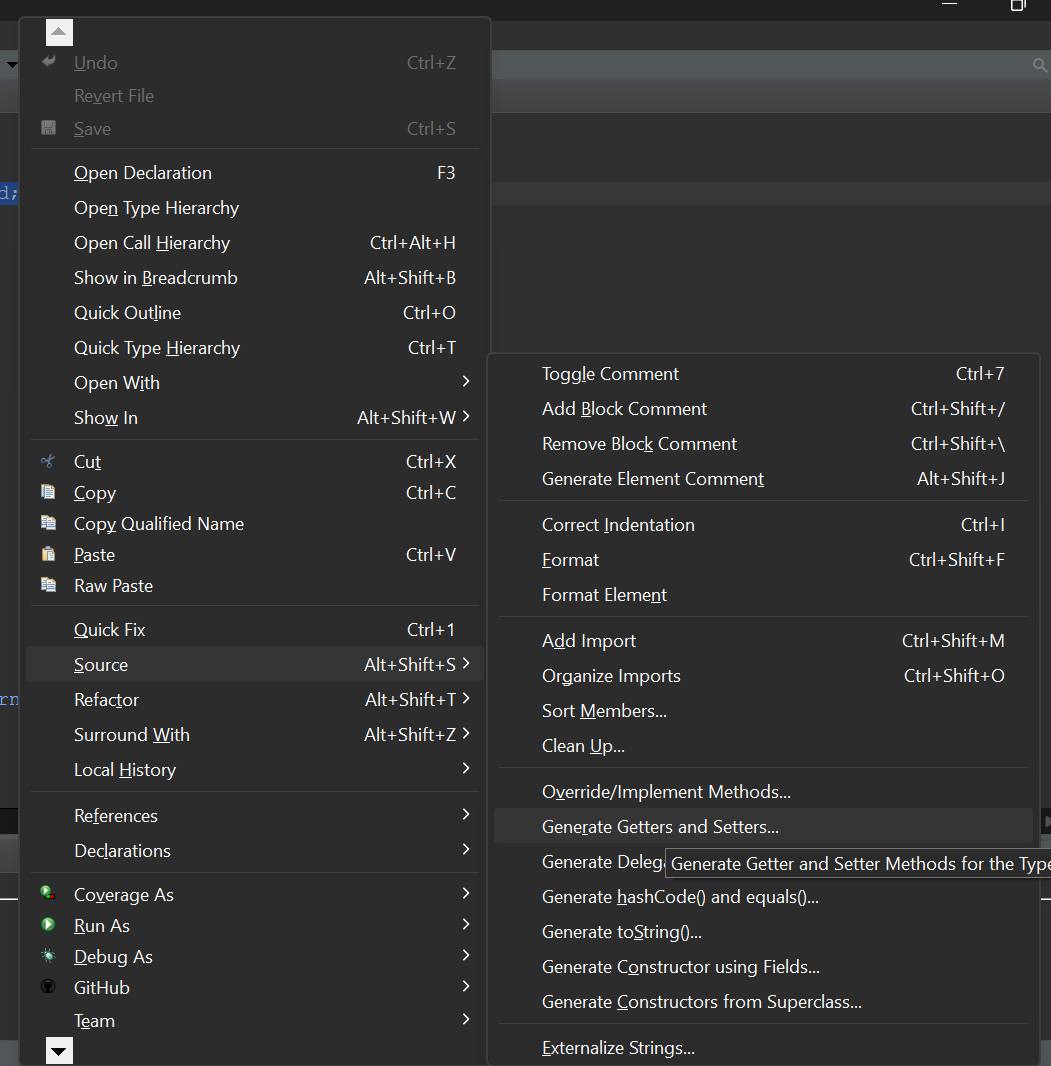
package model;

public class User {

String id, nama, username, password;

}

1. Tambahkan getter setter untuk masing-masing atribut. Untuk ini, bisa menggunakan pintasan dengan klik kanan pada laman source > Source > Generate Getters and Setters.

****

1. Setelah itu, getter dan setter akan ditambahkan secara otomatis.

package model;

public class User {

String id, nama, username, password;

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;

}

}

1. Tambahkan class lain seperti Customer, Order dan class Service. Masing-masing class diberi atribut dan getter setter nya.

* Class Customer

package model;

public class Customer {

String id,nama,alamat,noHP;

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 getAlamat() {

return alamat;

}

public void setAlamat(String alamat) {

this.alamat = alamat;

}

public String getNoHP() {

return noHP;

}

public void setNoHP(String noHP) {

this.noHP = noHP;

}

}

* Class Order

package model;

public class Order {

String id, id\_customer, id\_service,

tanggalOrder, tanggalSelesai, status\_bayar, status\_pesanan, id\_user;

int total;

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public String getId\_customer() {

return id\_customer;

}

public void setId\_customer(String id\_customer) {

this.id\_customer = id\_customer;

}

public String getId\_service() {

return id\_service;

}

public void setId\_service(String id\_service) {

this.id\_service = id\_service;

}

public String getTanggalOrder() {

return tanggalOrder;

}

public void setTanggalOrder(String tanggalOrder) {

this.tanggalOrder = tanggalOrder;

}

public String getTanggalSelesai() {

return tanggalSelesai;

}

public void setTanggalSelesai(String tanggalSelesai) {

this.tanggalSelesai = tanggalSelesai;

}

public String getStatus\_bayar() {

return status\_bayar;

}

public void setStatus\_bayar(String status\_bayar) {

this.status\_bayar = status\_bayar;

}

public String getStatus\_pesanan() {

return status\_pesanan;

}

public void setStatus\_pesanan(String status\_pesanan) {

this.status\_pesanan = status\_pesanan;

}

public String getId\_user() {

return id\_user;

}

public void setId\_user(String id\_user) {

this.id\_user = id\_user;

}

public int getTotal() {

return total;

}

public void setTotal(int total) {

this.total = total;

}

}

* Class Service

package model;

public class Service {

String id, jenis, satuan, status;

int harga;

public String getId() {

return id;

}

public void setId(String id) {

this.id = id;

}

public String getJenis() {

return jenis;

}

public void setJenis(String jenis) {

this.jenis = jenis;

}

public String getSatuan() {

return satuan;

}

public void setSatuan(String satuan) {

this.satuan = satuan;

}

public String getStatus() {

return status;

}

public void setStatus(String status) {

this.status = status;

}

public int getHarga() {

return harga;

}

public void setHarga(int harga) {

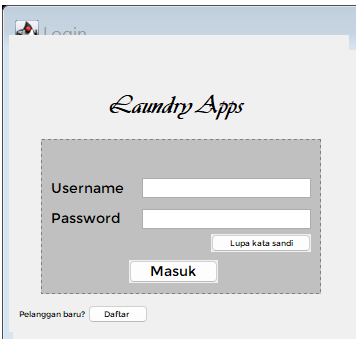
this.harga = harga;

}

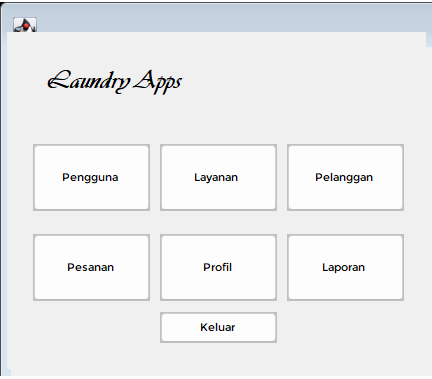
}

1. Selanjutnya, buat GUI menggunakan JFrame. GUI yang dibuat adalah LoginFrame dan MainFrame.

* LoginFrame



* MainFrame



1. Tambahkan method login pada class Users

public static boolean login(String username, String password) {

boolean isLogin = false;

User user = new User();

user.setId("1");

user.setNama("ahda");

user.setUsername("ahda");

user.setPassword("12345");

if(user.getUsername().equalsIgnoreCase(username.trim())

&& user.getPassword().equals(password)) {

isLogin = true;

} else {isLogin=false;}

return isLogin;

}

Method tersebut menggunakan getter setter yang telah dibuat pada class User. equalsIgnoreCase merupakan fungsi untuk mengembalikan String yang menjadi instance tanpa melihat aturan case String tersebut. trim() berfungsi untuk menghapus karakter spasi pada String.

1. Hubungkan LoginFrame dengan class User. Pada source tombol Masuk pada LoginFrame, tambahkan kode berikut:

btnNewButton\_1.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent e) {

if(User.*login*(textUsername.getText(),

passwordField.~~getText~~())) {

MainFrame main = new MainFrame();

main.setVisible(true);

dispose();

} else {

JOptionPane.*showMessageDialog*(null, "Gagal Login"); }

}

});

Method login dari class User digunakan, dengan instance adalah isi dari text field username dan password. setVisible(true) untuk menampilkan JFrame MainFrame, dan dispose() untuk menutup halaman ini, yaitu LoginFrame.

1. **Kesimpulan Praktikum**

Pada pemrograman berorientasi objek terdapat class, objek, atribut, method dan constructor. Untuk membangun sebuah aplikasi laundry digunakan dua package untuk memisahkan antara model dan gui. Pada model terdapat class User, Customer, Order, Service dan class lain sesuai kebutuhan. Pada ui terdapat JFrame LoginFrame dan MainFrame serta GUI lainnya nantinya yang mungkin dibutuhkan untuk membangun aplikasi Laundry.