

TUGAS PERTEMUAN 1

PRAKTIKUM PBO

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



Disusun Oleh :

Yusuf Fatha Mubina Alexander (2211533017)

Departemen Informatika

Fakultas Teknologi Informasi

Universitas Andalas

2024

A. Pendahuluan

1. Class

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.

2. Object

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.

3. Method

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.
- 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.
 - Accessor Method digunakan untuk membaca instance, menggunakan kata kunci `get` atau disebut `getter`.

- Mutator Method digunakan untuk membaca dan mengubah nilai, menggunakan kata kunci set atau disebut juga dengan setter dan akan mengembalikan value yang bersifat private.
 - 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().
4. Constructor
- Constructor yaitu blok kode yang mirip dengan method, menggunakan kata kunci new Ketika akan membuat sebuah object, nama constructor harus sama dengan nama class.

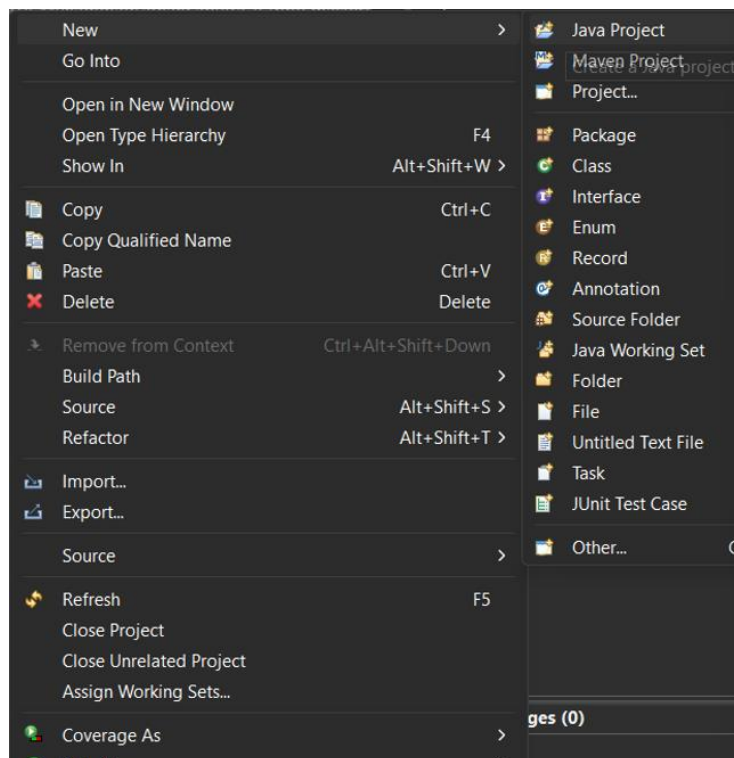
B. Tujuan

Tujuan dilakukannya praktikum ini adalah sebagai berikut:

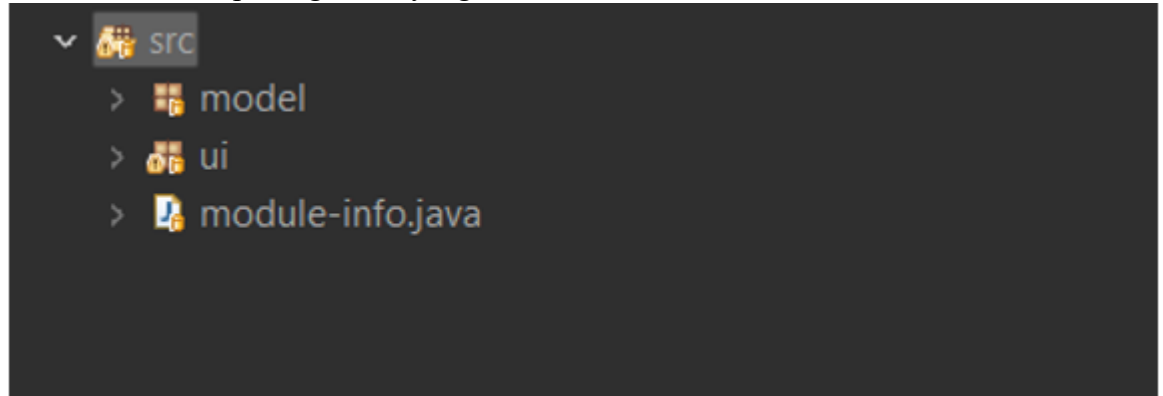
1. Membuat class, object, encapsulation, constructor dan method untuk aplikasi laundry seperti class User, Costumer, Service dan Order.
2. Membuat desain antarmuka aplikasi Laundry Login dan Halaman Utama.
3. Mampu menggunakan method pada JFrame/Tampilan aplikasi.
4. Mampu membuat fungsi berpindah dari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

C. Langkah Kerja Praktikum

1. Buka aplikasi eclipse, lalu buatlah project baru dengan nama laundryapps.



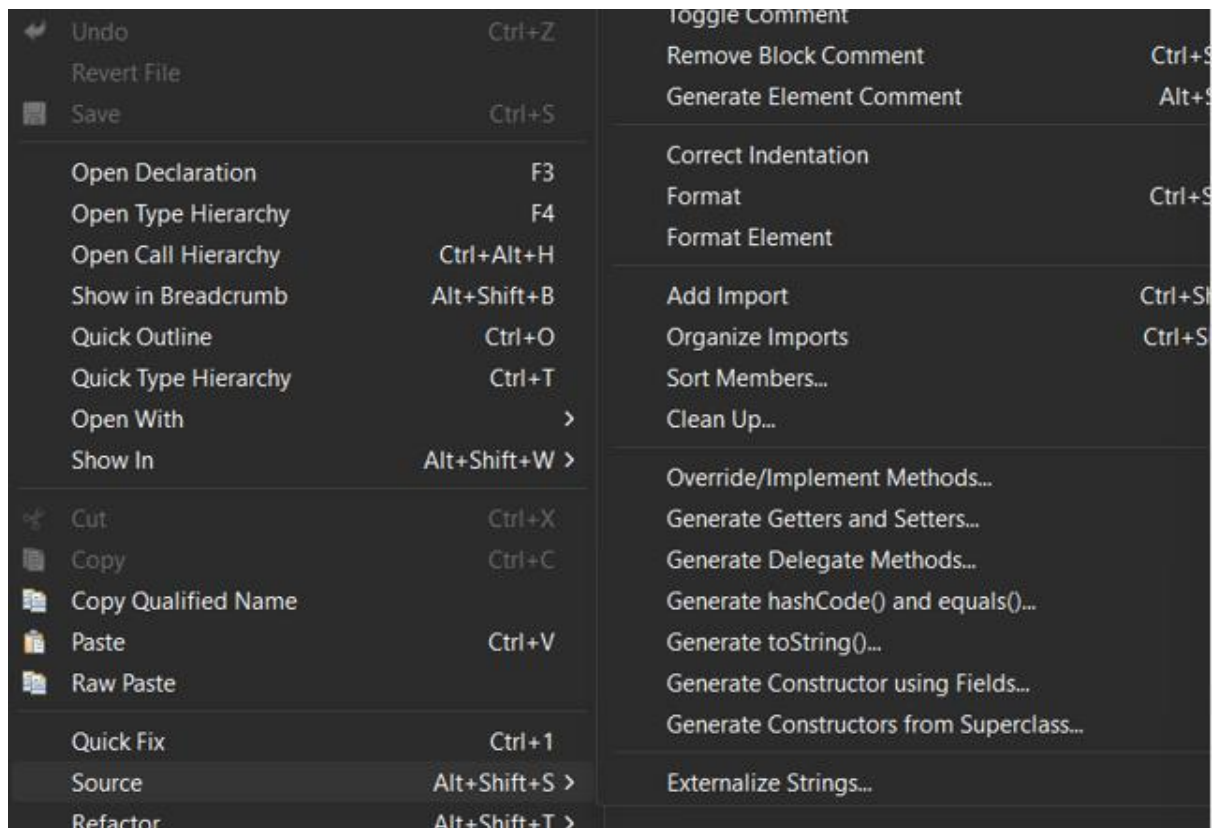
2. Kemudian buat 2 package baru yang Bernama “model” dan “ui”.



3. Lalu, buatlah class baru pada package model bernama user. Berilah Attribute di class tersebut dengan id, nama, username, password dengan tipe data string.

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

4. Dari attribute yang telah dibuat, buatlah setter dan getter untuk usernya dengan cara menekan tombol source di project, kemudian pilih sesuai attribute yang tersedia.



Select getters and setters to create:

> <input checked="" type="checkbox"/> * id	Select All
> <input checked="" type="checkbox"/> * nama	Deselect All
> <input checked="" type="checkbox"/> * password	Select Getters
> <input checked="" type="checkbox"/> * username	Select Setters

☐ Allow setters for `final` fields (remove 'final' modifier from fields if necessary)

Insertion point:
After 'password' ▼

Sort by:
Fields in getter/setter pairs ▼

Access modifier

<input checked="" type="radio"/> public	<input type="radio"/> protected	<input type="radio"/> package	<input type="radio"/> private
<input type="checkbox"/> final	<input type="checkbox"/> synchronized		

☐ Generate method comments

The format of the getters/setters may be configured on the [Code Templates](#) preference page.

8 of 8 selected.

```
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;
}
```

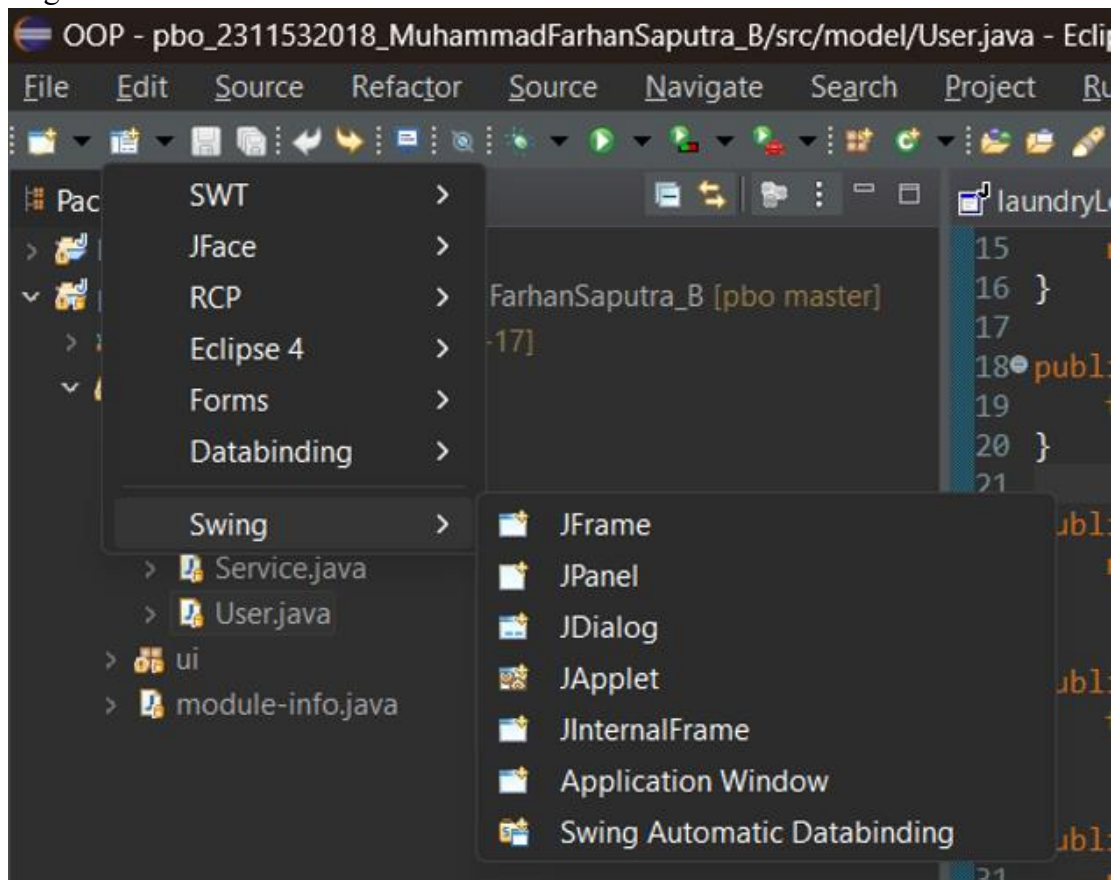
5. Buatlah login aplikasi dengan membuat method bernama “login”.

```
public static boolean login(String username, String password) {
    boolean isLogin = false;
    User user = new User();
    user.setId("1");
    user.setNama("fulan");
    user.setUsername("fulan");
    user.setPassword("12345");

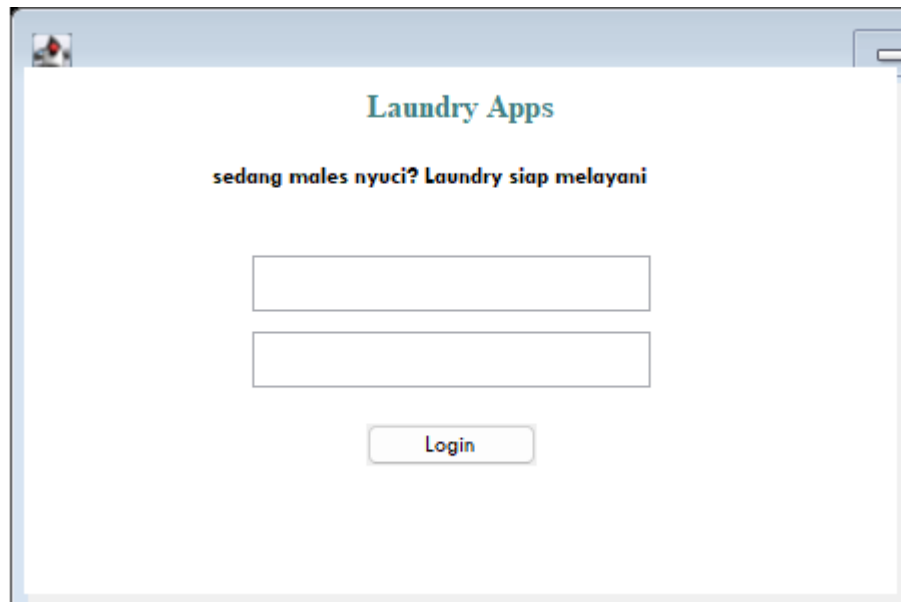
    String usernames = user.getUsername().trim();

    if(usernames.equalsIgnoreCase(username)
        && user.getPassword().equalsIgnoreCase(password)) {
        isLogin = true;
    } else {
        isLogin = false;
    }
    return isLogin;
}
```

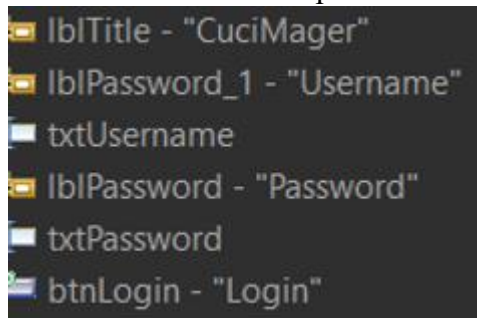
6. Selanjutnya, buatlah tampilan ui dengan JFrame di package ui dengan nama LoginFrame.



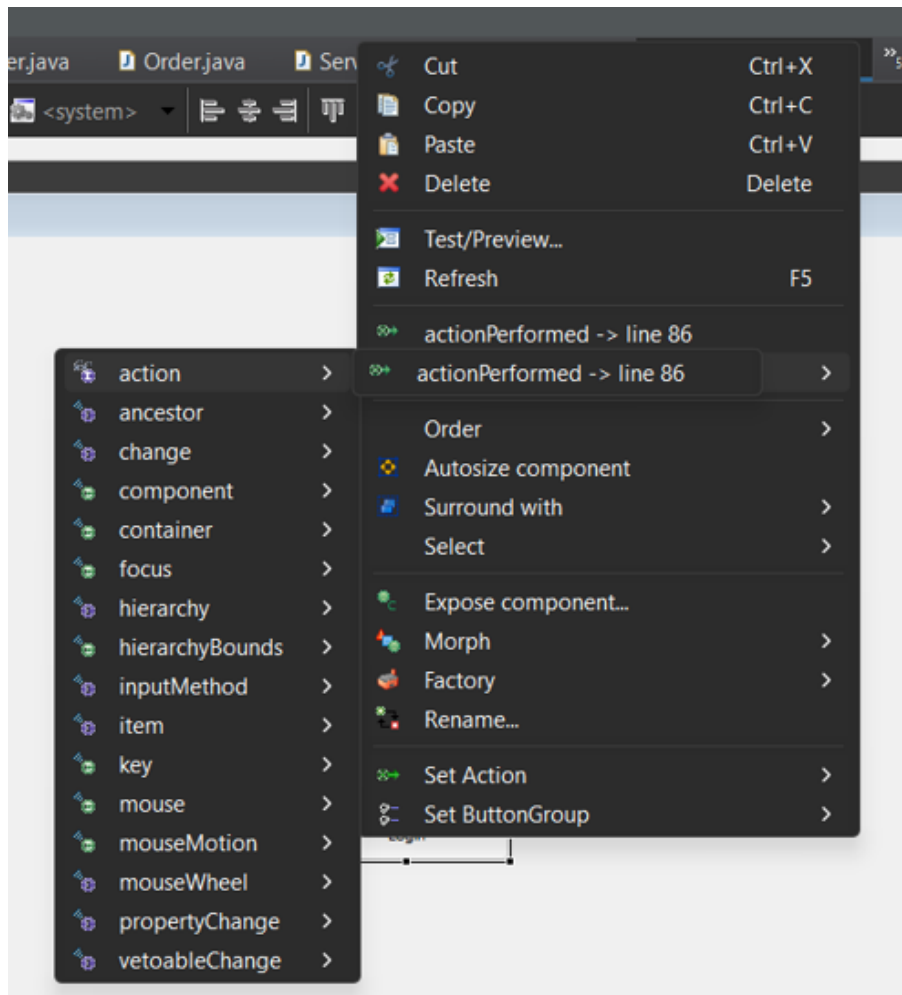
7. Setelah membuat LoginFrame, maka akan muncul tampilan seperti ini.



8. Beri nama variabel komponen sesuai ketentuan.



9. Pada bagian tombol, klik kanan kemudian pilih add even handler > action > actionPerformed.



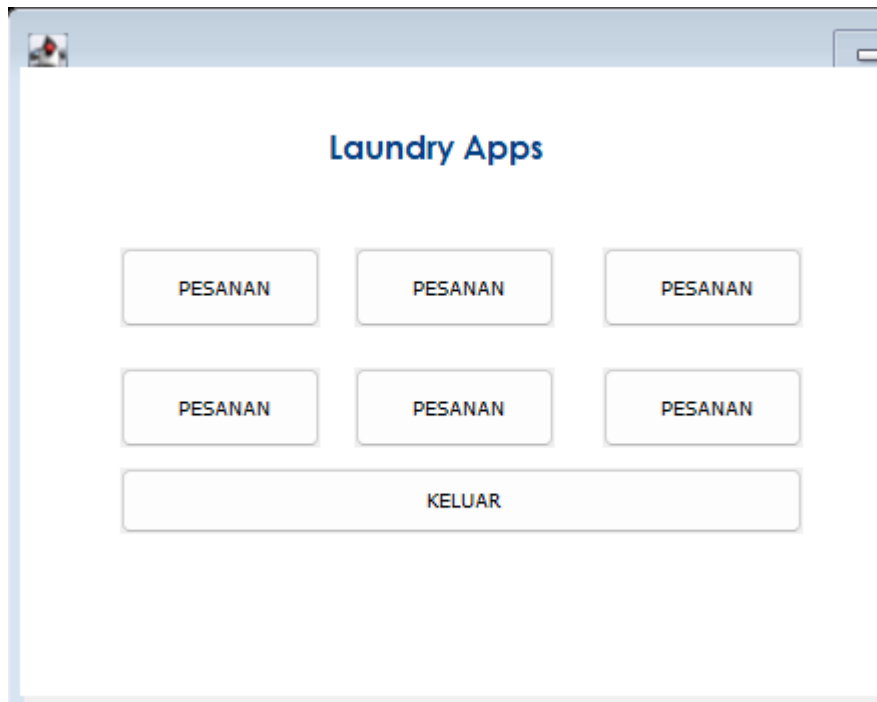
Kemudian panggil method login pada class User dengan mengirimkan parameter yang diambil dari txtUsername dan txtPassword, jika cocok maka akan tampil halaman utama.

```

JButton btnLogin = new JButton("Login");
btnLogin.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        if(User.login(txtUsername.getText().trim(), txtPassword.getText().trim())) {
            new MainFrame().setVisible(true);
            dispose();
        } else {
            JOptionPane.showMessageDialog(null, "Login Gagal");
        }
    }
});
btnLogin.setBounds(326, 379, 138, 36);
contentPane.add(btnLogin);

```

10. Buatlah sebuah JFrame baru di package ui dengan nama MainFrame



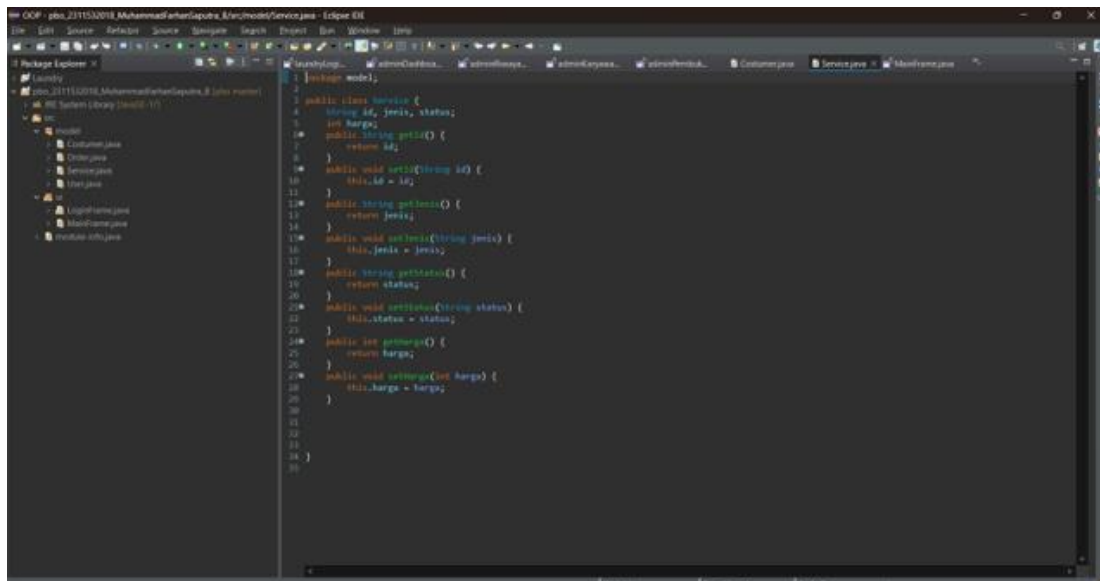
11. Buatlah class dengan nama Costumer dengan attribute id, nama, Alamat dan nomor hp, buatkan setter dan getter untuk menambahkan object Costumer.

```

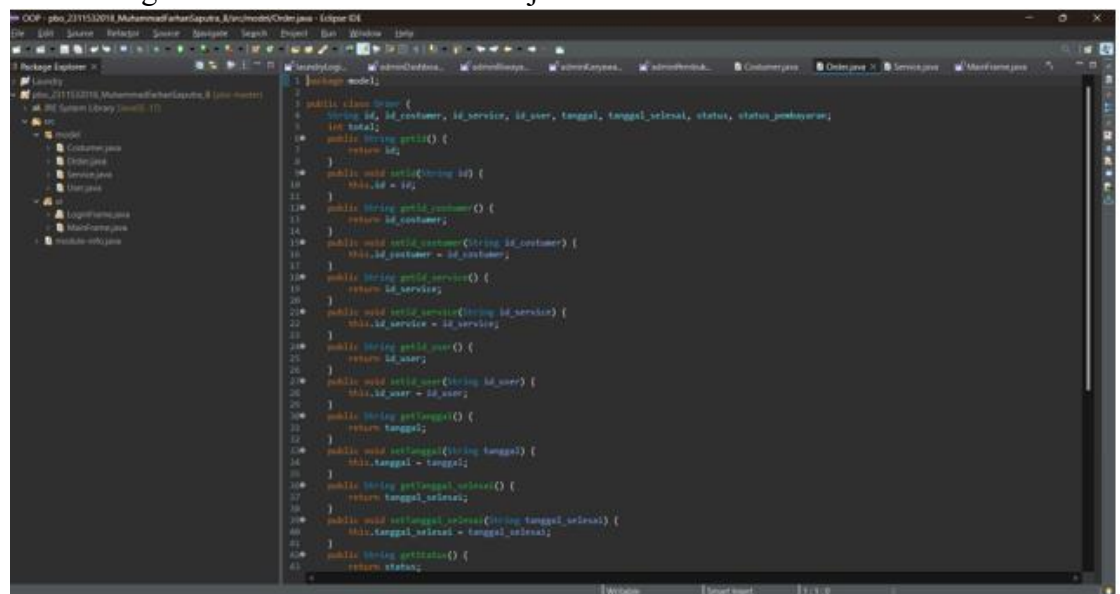
1 package model;
2
3 public class Customer {
4     String id, nama, alamat, nohp;
5
6     public String getId() {
7         return id;
8     }
9
10    public void setId(String id) {
11        this.id = id;
12    }
13
14    public String getNama() {
15        return nama;
16    }
17
18    public void 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 getNohp() {
31        return nohp;
32    }
33
34    public void setNohp(String nohp) {
35        this.nohp = nohp;
36    }
37
38
39
40 }

```

12. Buatlah class dengan nama Service dengan attribute id, jenis, harga dan status, buatkan setter dan getter untuk menambahkan object Service.



13. Buatlah class dengan nama Order dengan attribute id, id_costumer, id_service, id_user, total, tanggal, tanggal_selesai, status, status_pembayaran , buatkan setter dan getter untuk menambahkan object Order.



D. Kesimpulan

Class dalam Java adalah blueprint untuk objek yang memiliki sifat dan atribut serupa, seperti class Mahasiswa untuk objek mahasiswa seperti fulan. Objek merupakan representasi entitas dunia nyata dengan state (atribut), behavior (method), dan identity (nama unik). Method adalah blok kode yang dapat dipanggil berulang kali, terdiri dari predefined method (metode bawaan Java), user-defined method (metode yang dibuat programmer), static method (yang dapat dipanggil tanpa objek), instance method (yang memerlukan objek, termasuk accessor dan mutator), abstract method (tanpa body dalam class abstract), dan factory method (mengembalikan objek ke class). Constructor adalah blok kode yang digunakan untuk membuat objek dengan nama yang sama dengan class dan menggunakan kata kunci new.