

ACTIVITY PERTEMUAN 2

NAMA : Muhammad Rifqi Dwi Putra
NPM : 51422145
KELAS : 4IA08
MATERI : Anatomi class, struktur aplikasi java, object oriented programming (OOP) dengan java
MATA PRAKTIKUM : Rekayasa Perangkat Lunak 2

(Screenshoot langkah-langkah sesuai video pembelajaran dan jelaskan dengan ringkas)

1. Jelaskan anatomi dasar dari sebuah class dalam Java. Apa saja komponen utama yang ada dalam class dan bagaimana fungsinya? Berikan contoh sederhana dari sebuah class dengan atribut dan method didalamnya

Class adalah blueprint untuk membuat objek. Di dalamnya terdapat data (atribut) dan perilaku (method) yang membentuk karakter dan fungsi dari objek.

Komponen utama class dan fungsinya :

- Atribut: menyimpan data objek
- Method: menjalankan aksi atau logika
- Constructor: menginisialisasi objek
- Access modifier: mengatur hak akses (public, private, dll)

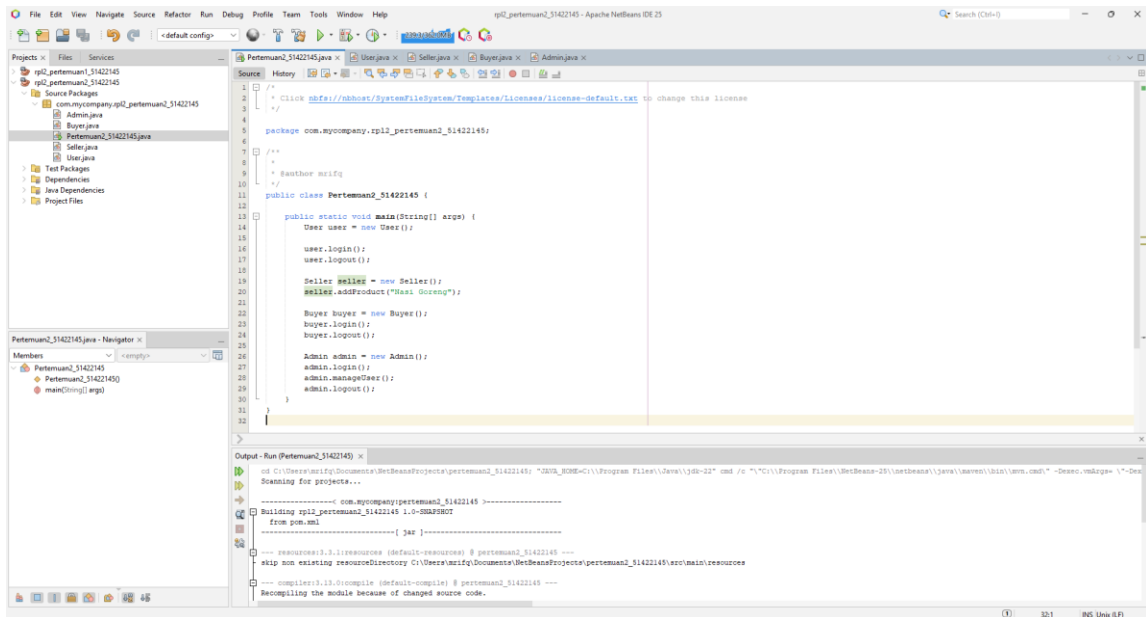
Contoh nya :

```
public class Mobil {  
    String merk;  
    int tahun;  
  
    public Mobil(String merk, int tahun) {  
        this.merk = merk;  
        this.tahun = tahun;  
    }  
  
    public void nyalakanMesin() {  
        System.out.println("Mesin " + merk + " dinyalakan.");  
    }  
}
```

Class Mobil memiliki dua atribut (merk dan tahun), satu constructor untuk menginisialisasi nilai atribut saat objek dibuat, dan satu method (nyalakanMesin) yang mencetak informasi ke layar. Dengan class ini, kita bisa membuat objek Mobil dan memanggil fungsinya sesuai kebutuhan.

2. Screenshot kode, output, dan beri penjelasan singkat dari program yang dibuat pada pertemuan kedua

File Pertemuan2_51422145.java :



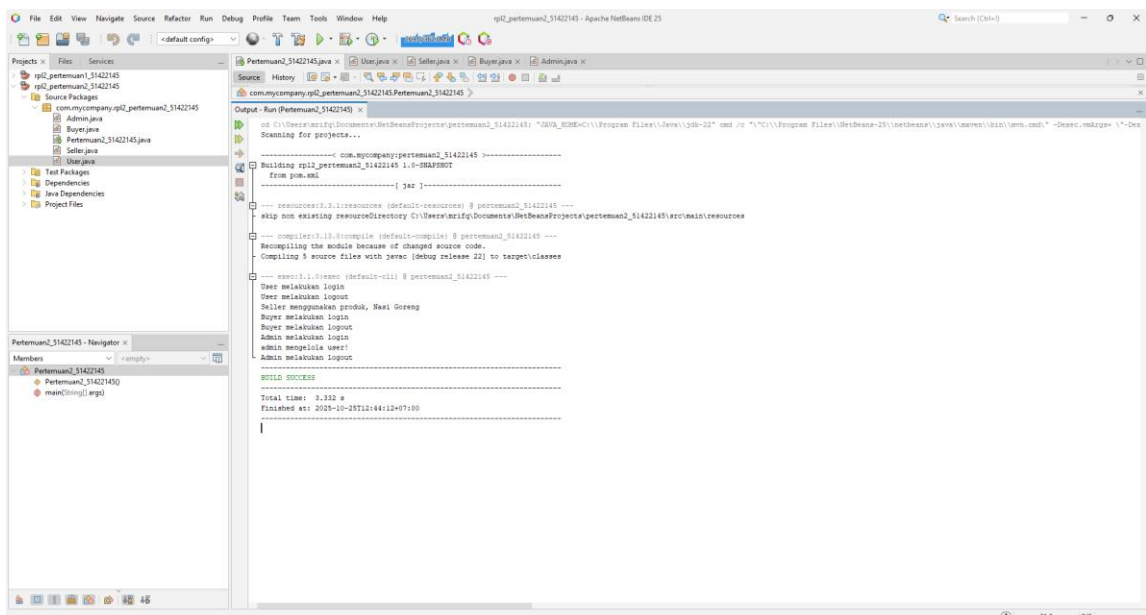
```
1  /*
2  * Click https://pbbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3  */
4
5  package com.mycompany.rpl2_pertemuan2_51422145;
6
7  /**
8   *
9   * @author xxiifq
10  */
11  public class Pertemuan2_51422145 {
12
13      public static void main(String[] args) {
14          User user = new User();
15
16          user.login();
17          user.logout();
18
19          Seller seller = new Seller();
20          seller.addProduct("Nasi Goreng");
21
22          Buyer buyer = new Buyer();
23          buyer.login();
24          buyer.logout();
25
26          Admin admin = new Admin();
27          admin.login();
28          admin.manageUser();
29          admin.logout();
30      }
31  }
32  }
```

Output - Run (Pertemuan2_51422145) x

```
cd C:\Users\xxiifq\Documents\NetBeansProjects\pertemuan2_51422145; "C:\WINDOWS\system32\cmd.exe" /s /c "%C:\Program Files\NetBeans-23\netbeans\java\bin\javac.exe" -Denc.mdirger "%-Des
Scanning for projects...
----- com.mycompany:pertemuan2_51422145 -----
Building pertemuan2_51422145 1.0-SNAPSHOT
from pom.xml
----- [ JAR ] -----
----- resources:3.3.1:resources (default-resources) @ pertemuan2_51422145 -----
skip non existing resourceDirectory C:\Users\xxiifq\Documents\NetBeansProjects\pertemuan2_51422145\src\main\resources
----- compiler:3.11.0:compile (default-compile) @ pertemuan2_51422145 -----
Recompiling the module because of changed source code.
```

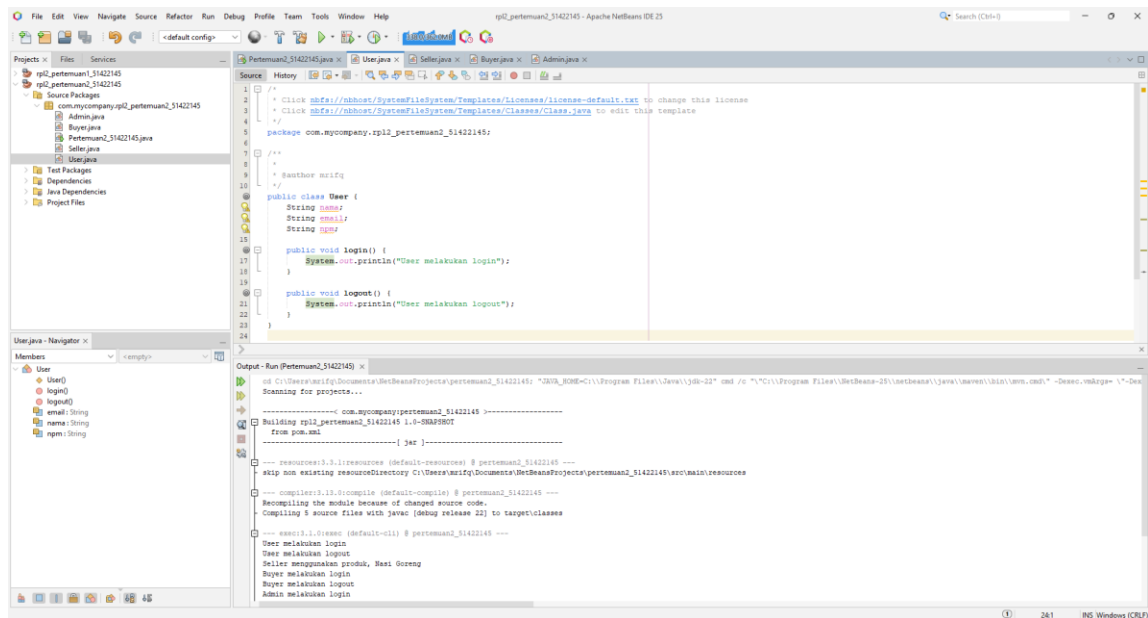
Kode program di atas merupakan contoh penerapan pemrograman berorientasi objek (OOP) di Java. Program ini memiliki class utama Pertemuan2_51422145 yang berisi method main() sebagai titik awal eksekusi. Di dalamnya dibuat beberapa objek dari class User, Seller, Buyer, dan Admin. Setiap objek menjalankan method sesuai perannya, seperti login(), logout(), addProduct("Nasi Goreng"), dan manageUser(). Melalui kode ini terlihat bahwa setiap class memiliki tanggung jawab dan perilaku berbeda, namun tetap saling terhubung melalui konsep inheritance dan prinsip OOP.

Output Pertemuan2_51422145.java :



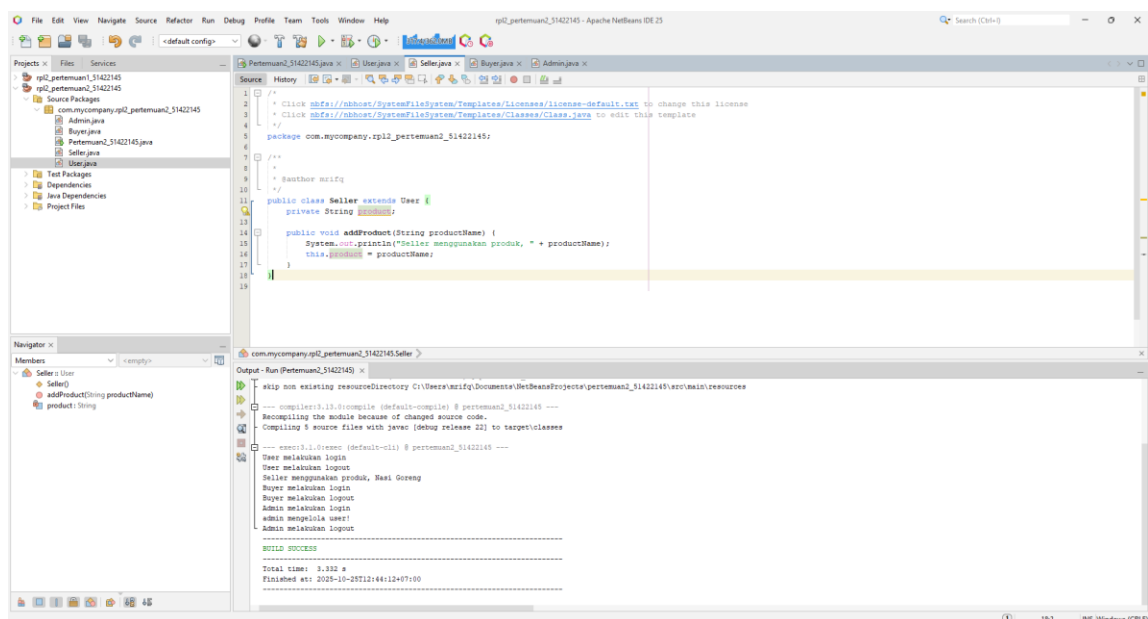
```
cd C:\Users\xxiifq\Documents\NetBeansProjects\pertemuan2_51422145; "C:\WINDOWS\system32\cmd.exe" /s /c "%C:\Program Files\NetBeans-23\netbeans\java\bin\javac.exe" -Denc.mdirger "%-Des
Scanning for projects...
----- com.mycompany:pertemuan2_51422145 -----
Building pertemuan2_51422145 1.0-SNAPSHOT
from pom.xml
----- [ JAR ] -----
----- resources:3.3.1:resources (default-resources) @ pertemuan2_51422145 -----
skip non existing resourceDirectory C:\Users\xxiifq\Documents\NetBeansProjects\pertemuan2_51422145\src\main\resources
----- compiler:3.11.0:compile (default-compile) @ pertemuan2_51422145 -----
Recompiling the module because of changed source code.
----- compiler:3.11.0:compile (default-compile) @ pertemuan2_51422145 -----
Compiling 3 source files with javac [debug release 22] to target\classes
----- main:1.0:run (default-cli) @ pertemuan2_51422145 -----
User melakukan login
User melakukan logout
Seller menggunakan produk, Nasi Goreng
Buyer melakukan login
Buyer melakukan logout
Admin melakukan login
Admin mengelola user!
Admin melakukan logout
BUILD SUCCESS
Total time: 3.332 s
Finished at: 2025-10-20T12:44:12+07:00
1
```

File User.java :



Kode di atas mendefinisikan sebuah **class** bernama User yang merepresentasikan data dan perilaku dasar dari seorang pengguna. Class ini memiliki tiga **atribut**: nama, email, dan npm, yang berfungsi untuk menyimpan informasi pengguna. Selain itu, terdapat dua **method**, yaitu login() dan logout(), yang menampilkan pesan ke konsol saat pengguna melakukan proses masuk dan keluar sistem. Class ini menjadi dasar yang dapat diturunkan (inheritance) oleh class lain seperti Seller, Buyer, atau Admin, agar mereka dapat mewarisi fungsi umum dari User.

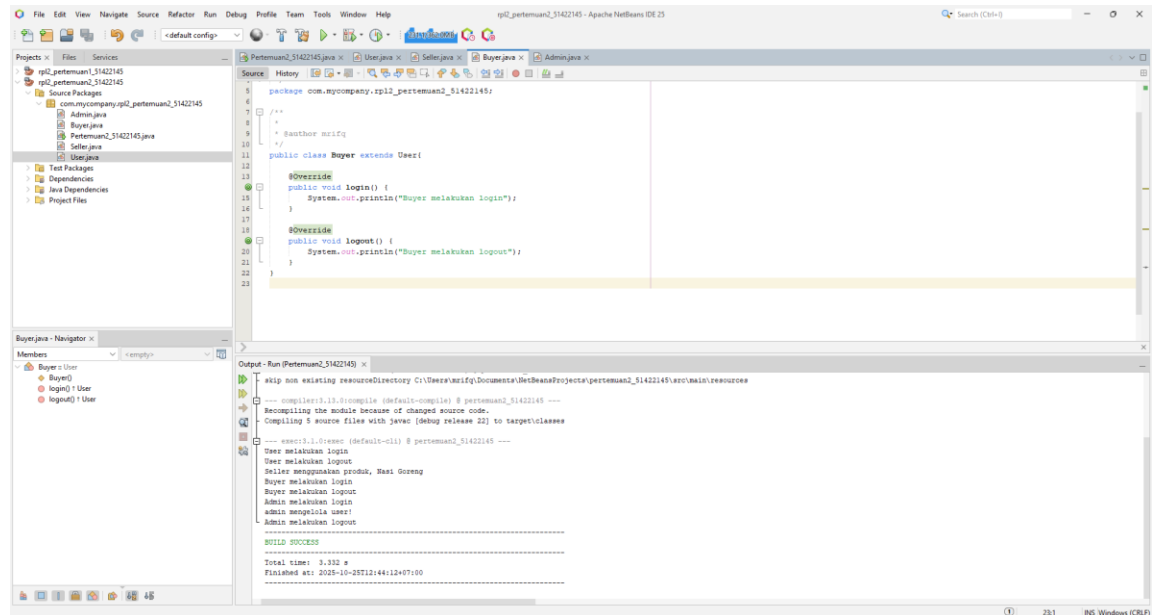
File Seller.java :



Kode di atas menunjukkan bahwa Seller merupakan subclass dari User yang mewarisi atribut dan method seperti nama, email, npm, login(), dan logout(). Class ini menambahkan atribut product serta method addProduct(String productName) untuk menambah dan menampilkan produk yang

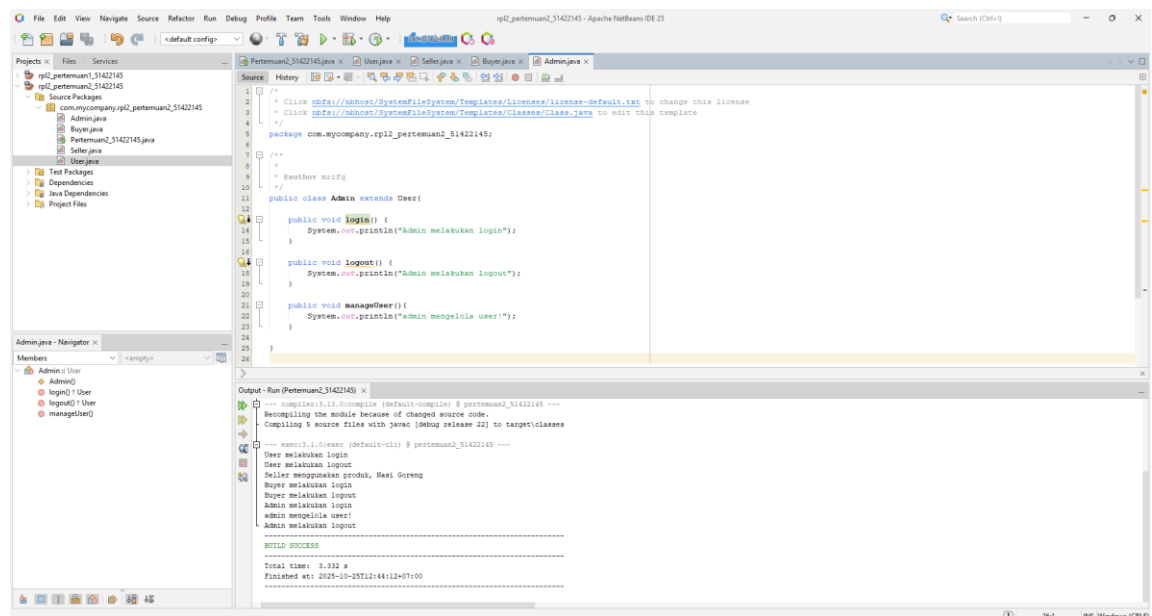
dimiliki penjual. Secara keseluruhan, Seller memperluas fungsi User dengan kemampuan khusus dalam mengelola produk.

File Buyer.java :



Kode di atas menunjukkan bahwa Buyer adalah subclass dari User yang menggunakan anotasi `@Override` untuk mengubah (menimpa) perilaku method `login()` dan `logout()` yang diwarisi dari User. Hal ini membuat Buyer memiliki pesan khusus saat login dan logout, sesuai dengan perannya sebagai pembeli dalam sistem.

File Admin.java :



Kode di atas menunjukkan bahwa Admin adalah subclass dari User yang menerima method login() dan logout() agar menampilkan pesan khusus untuk admin. Selain itu, class ini memiliki method tambahan bernama manageUser() yang digunakan untuk menggambarkan kemampuan admin dalam mengelola pengguna lain. Dengan demikian, Admin memperluas fungsi User dengan peran khusus sebagai pengelola sistem.