

LAPORAN PRAKTIKUM MINGGU 2

CLASS DAN OBJECT

NAMA : ALVIRA LIBRA RAMADHANI

KELAS : 3IKRA

NIM : 240202851

1. TUJUAN

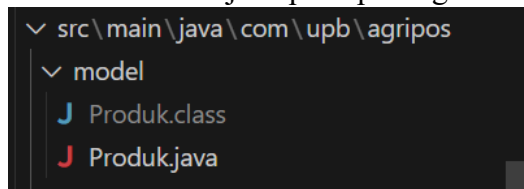
- Mahasiswa mampu menjelaskan konsep class, object, atribut, dan method dalam OOP.
- Mahasiswa mampu menerapkan access modifier dan enkapsulasi dalam pembuatan class.
- Mahasiswa mampu mengimplementasikan class Produk pertanian dengan atribut dan method yang sesuai.
- Mahasiswa mampu mendemonstrasikan instansiasi object serta menampilkan data produk pertanian di console.
- Mahasiswa mampu menyusun laporan praktikum dengan bukti kode, hasil eksekusi, dan analisis sederhana.

2. DASAR TEORI

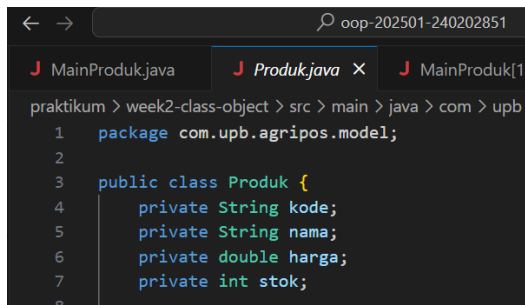
- Class adalah blueprint atau cetak biru dari objek yang berisi atribut dan metode.
- Object merupakan instansiasi dari class yang memiliki data nyata.
- Enkapsulasi adalah konsep menyembunyikan data (atribut) agar hanya bisa diakses melalui metode tertentu (getter dan setter).
- Method static dapat dipanggil tanpa membuat objek dari class tersebut.
- Pemisahan package (model, util) membantu menjaga struktur dan kerapian proyek.

3. LANGKAH PRAKTIKUM

- Membuat Class Produk
 - Buat file Produk.java pada package model



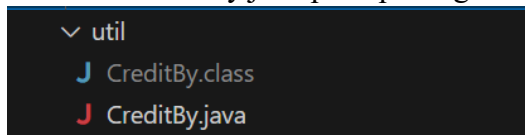
- Tambahkan atribut: kode, nama, harga, dan stok.



```
← → oop-202501-240202851
MainProduk.java Produk.java MainProduk[1]
praktikum > week2-class-object > src > main > java > com > upb
1 package com.upb.agripos.model;
2
3 public class Produk {
4     private String kode;
5     private String nama;
6     private double harga;
7     private int stok;
8 }
```

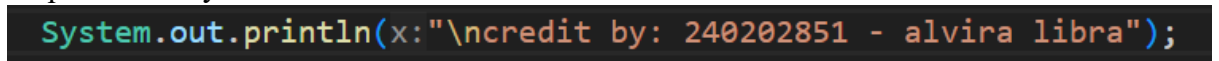
b. Membuat Class CreditBy

1. Buat file CreditBy.java pada package util.



```
util
  CreditBy.class
  CreditBy.java
```

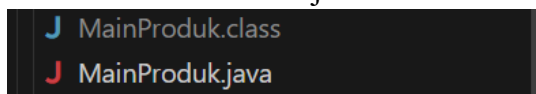
2. Isi class dengan method statis untuk menampilkan identitas mahasiswa di akhir output: credit by: <NIM> - <NAMA>



```
System.out.println(x:"\ncredit by: 240202851 - alvira libra");
```

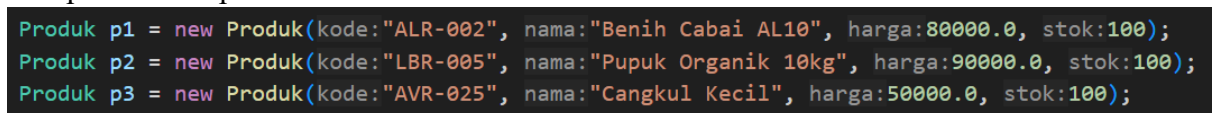
c. Membuat Objek Produk dan Menampilkan Credit

1. Buat file MainProduk.java.



```
MainProduk.class
MainProduk.java
```

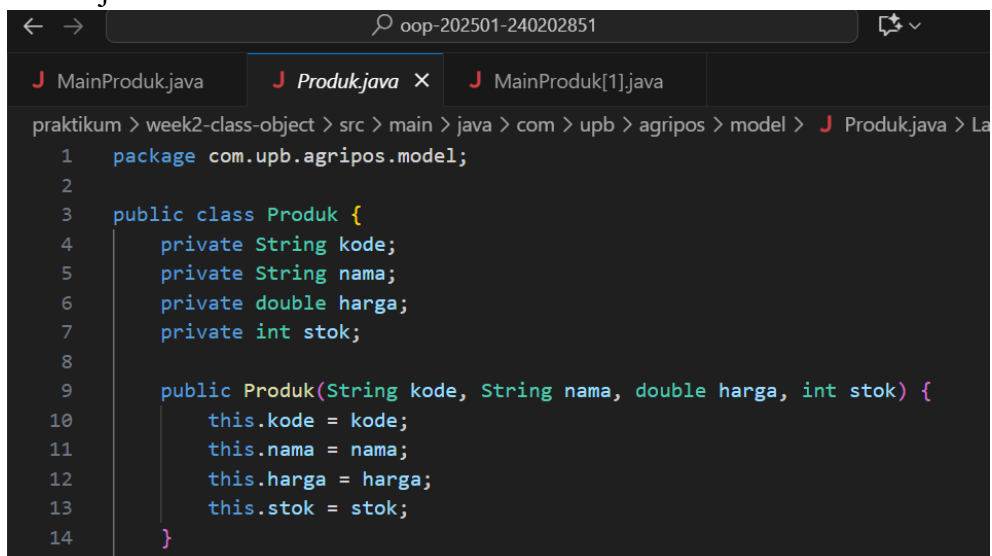
2. Instansiasi minimal tiga objek produk, misalnya “Benih Padi”, “Pupuk Urea”, dan satu produk alat pertanian.



```
Produk p1 = new Produk(kode:"ALR-002", nama:"Benih Cabai AL10", harga:80000.0, stok:100);
Produk p2 = new Produk(kode:"LBR-005", nama:"Pupuk Organik 10kg", harga:90000.0, stok:100);
Produk p3 = new Produk(kode:"AVR-025", nama:"Cangkul Kecil", harga:50000.0, stok:100);
```

4. KODE PROGRAM

a. Produk.java



```
← → oop-202501-240202851
MainProduk.java Produk.java MainProduk[1].java
praktikum > week2-class-object > src > main > java > com > upb > agripos > model > Produk.java > La
1 package com.upb.agripos.model;
2
3 public class Produk {
4     private String kode;
5     private String nama;
6     private double harga;
7     private int stok;
8
9     public Produk(String kode, String nama, double harga, int stok) {
10         this.kode = kode;
11         this.nama = nama;
12         this.harga = harga;
13         this.stok = stok;
14     }
15 }
```

```

16 public String getCode() { return kode; }
17 public void setCode(String kode) { this.kode = kode; }
18
19 public String getName() { return nama; }
20 public void setName(String nama) { this.nama = nama; }
21
22 public double getHarga() { return harga; }
23 public void setHarga(double harga) { this.harga = harga; }
24
25 public int getStok() { return stok; }
26 public void setStok(int stok) { this.stok = stok; }
27

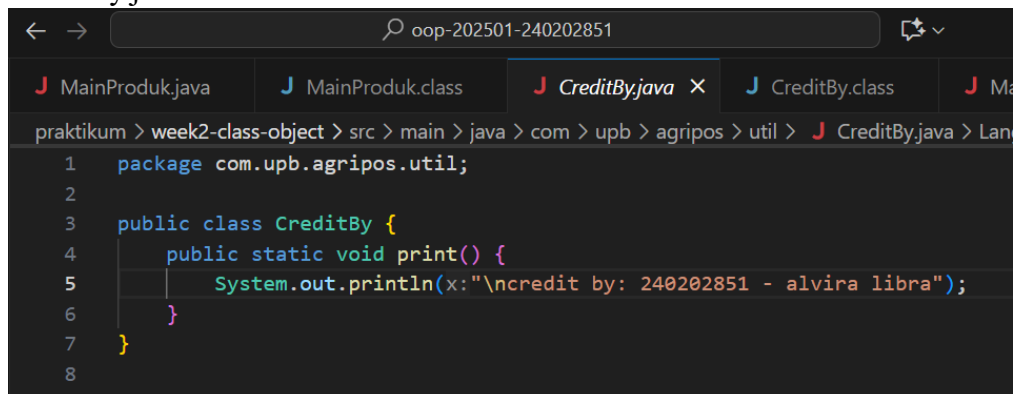
```

```

25 public int getStok() { return stok; }
26 public void setStok(int stok) { this.stok = stok; }
27
28 public void tambahStok(int jumlah) {
29     stok += jumlah;
30     System.out.println("Berhasil menambah stok " + jumlah + " unit. Stok sekarang: " + stok);
31 }
32
33 public void kurangiStok(int jumlah) {
34     if (stok >= jumlah) {
35         stok -= jumlah;
36         System.out.println("Berhasil mengurangi stok " + jumlah + " unit. Stok sekarang: " + stok);
37     } else {
38         System.out.println("Stok tidak cukup! Tidak bisa mengurangi " + jumlah + " unit.");
39     }
40 }
41
42 public void tampilkanInfo() {
43     System.out.println("Kode: " + kode + ", Nama: " + nama +
44         ", Harga: " + harga + ", Stok: " + stok);
45 }
46

```

b. CreditBy.java



```

← → oop-202501-240202851
MainProduk.java MainProduk.class CreditBy.java × CreditBy.class
praktikum > week2-class-object > src > main > java > com > upb > agripos > util > CreditBy.java > Lan
1 package com.upb.agripos.util;
2
3 public class CreditBy {
4     public static void print() {
5         System.out.println(x:"\ncredit by: 240202851 - alvira libra");
6     }
7 }
8

```

c. MainProduk.java

```
oop-202501-240202851
MainProduk.java x MainProduk.class Produk.java CreditBy.class MainProduk[1].java
praktikum > week2-class-object > src > main > java > com > upb > agripis > MainProduk.java > Language Support for Java(TM) by Red Hat
1 package com.upb.agripis;
2
3 import com.upb.agripis.model.Produk;
4 import com.upb.agripis.util.CreditBy;
5
6 public class MainProduk {
7     Run main | Debug main | Run | Debug
8     public static void main(String[] args) {
9
10         Produk p1 = new Produk(kode:"ALR-002", nama:"Benih Cabai AL10", harga:80000.0, stok:100);
11         Produk p2 = new Produk(kode:"LBR-005", nama:"Pupuk Organik 10kg", harga:90000.0, stok:100);
12         Produk p3 = new Produk(kode:"AVR-025", nama:"Cangkul Kecil", harga:50000.0, stok:100);
13
14         System.out.println(x:"=== Info Awal Produk ===");
15         p1.tampilkanInfo();
16         p2.tampilkanInfo();
17         p3.tampilkanInfo();
18
19         System.out.println(x:"\n=== Menambah Stok Produk ===");
20         System.out.println(x:"Menambah stok Benih Cabai AL10 sebanyak 50");
21         p1.tambahStok(jumlah:50);
22         p1.tampilkanInfo();
23
24         System.out.println(x:"\n=== Mengurangi Stok Produk ===");
25         System.out.println(x:"Mengurangi stok Cangkul Kecil sebanyak 50");
26         p3.kurangiStok(jumlah:50);
27         p3.tampilkanInfo();
28
29         CreditBy.print();
30     }
}
```

5. HASIL EKSEKUSI

- a. Gunakan perintah javac untuk mengompilasi dan java untuk menjalankan program seperti berikut:

```
PS C:\Users\LENOVO\oop-202501-240202851\praktikum\week2-class-object\src\main\java> javac com/upb/agripis/model/Produk.java
PS C:\Users\LENOVO\oop-202501-240202851\praktikum\week2-class-object\src\main\java> javac com/upb/agripis/util/CreditBy.java
PS C:\Users\LENOVO\oop-202501-240202851\praktikum\week2-class-object\src\main\java> javac com/upb/agripis/MainProduk.java
PS C:\Users\LENOVO\oop-202501-240202851\praktikum\week2-class-object\src\main\java> java com.upb.agripis.MainProduk
```

- b. Berikut tampilan hasil eksekusi program di terminal:

```
PS C:\Users\LENOVO\oop-202501-240202851\praktikum\week2-class-object\src\main\java> java com.upb.agripis.MainProduk
=== Info Awal Produk ===
Kode: ALR-002, Nama: Benih Cabai AL10, Harga: 80000.0, Stok: 100
Kode: LBR-005, Nama: Pupuk Organik 10kg, Harga: 90000.0, Stok: 100
Kode: AVR-025, Nama: Cangkul Kecil, Harga: 50000.0, Stok: 100

=== Menambah Stok Produk ===
Menambah stok Benih Cabai AL10 sebanyak 50
Berhasil menambah stok 50 unit. Stok sekarang: 150
Kode: ALR-002, Nama: Benih Cabai AL10, Harga: 80000.0, Stok: 150

=== Mengurangi Stok Produk ===
Mengurangi stok Cangkul Kecil sebanyak 50
Berhasil mengurangi stok 50 unit. Stok sekarang: 50
Kode: AVR-025, Nama: Cangkul Kecil, Harga: 50000.0, Stok: 50

credit by: 240202851 - alvira libra
PS C:\Users\LENOVO\oop-202501-240202851\praktikum\week2-class-object\src\main\java>
```

6. ANALISIS

- a. Jelaskan bagaimana kode berjalan.
Kode berjalan dengan konsep OOP. Class Produk menyimpan data dengan enkapsulasi (private + getter/setter), lalu di MainProduk dibuat beberapa objek produk dan ditampilkan lewat getter. Terakhir, CreditBy.print() menampilkan identitas mahasiswa.
- b. Apa perbedaan pendekatan minggu ini dibanding minggu sebelumnya.
Pendekatan minggu ini berbeda dari sebelumnya karena sudah memakai OOP, bukan prosedural, sehingga kode lebih rapi dan terstruktur.
- c. Kendala yang dihadapi dan cara mengatasinya.
Kendala yang muncul yaitu error struktur folder dan penamaan class. Solusinya dengan menyesuaikan package serta menjalankan javac dan java dari direktori yang benar.

7. KESIMPULAN

Melalui praktikum ini, mahasiswa memahami pembuatan class, instansiasi object, serta penerapan enkapsulasi. Dengan pendekatan OOP, kode menjadi lebih terstruktur, mudah dikembangkan, dan memisahkan tanggung jawab antar bagian program.

8. CHEKLIST KEBERHASILAN

- ☒ Class Produk berhasil dibuat dengan atribut dan method yang lengkap.
- ☒ Class CreditBy berhasil dibuat dan dipanggil di program utama.
- ☒ Objek produk berhasil diinstansiasi dan ditampilkan.
- ☒ Enkapsulasi sudah diterapkan dengan benar.
- ☒ Commit dengan pesan sesuai instruksi berhasil dilakukan.
- ☒ Screenshot hasil eksekusi telah dilampirkan.
- ☒ Laporan singkat telah dibuat.

9. QUIZ

- a. Mengapa atribut sebaiknya dideklarasikan sebagai private dalam class?
Jawaban: Agar data terlindungi dari akses langsung oleh class lain dan hanya bisa diubah melalui method khusus (getter dan setter).
- b. Apa fungsi getter dan setter dalam enkapsulasi?
Jawaban: Getter digunakan untuk mengambil nilai atribut, sedangkan setter digunakan untuk mengubah nilainya dengan cara yang terkontrol.
- c. Bagaimana cara class Produk mendukung pengembangan aplikasi POS yang lebih kompleks?
Jawaban: Class Produk menjadi dasar penyimpanan dan pengelolaan data barang, sehingga mudah dikembangkan untuk fitur seperti transaksi, stok, dan laporan penjualan.