

Laporan Jobsheet 2

Satria Wiguna/TI_1D/Absen 26

Percobaan 1: Deklarasi Class, Atribut dan Method

```
public class Buku26 {  
    String judul, pengarang;  
    int halaman, stok, harga;  
  
    void tampilkaninformasi(){  
        System.out.println("Judul : " + judul);  
        System.out.println("Pengarang : " + pengarang);  
        System.out.println("Jumlah halaman : " + halaman);  
        System.out.println("Sisa stok : " + stok);  
        System.out.println("Harga : RP " + harga);  
    }  
    void terjual(int jml){  
        stok -= jml;  
    }  
    void restock(int jml){  
        stok += jml;  
    }  
    void gantiharga(int hrg) {  
        harga = hrg;  
    }  
}
```

Pertanyaan

1. Sebutkan dua karakteristik class atau object!
2. Perhatikan class Buku pada Praktikum 1 tersebut, ada berapa atribut yang dimiliki oleh class Buku? Sebutkan apa saja atributnya!
3. Ada berapa method yang dimiliki oleh class tersebut? Sebutkan apa saja methodnya!
4. Perhatikan method terjual() yang terdapat di dalam class Buku. Modifikasi isi method tersebut sehingga proses pengurangan hanya dapat dilakukan jika stok masih ada (lebih besar dari 0)!
5. Menurut Anda, mengapa method restock() mempunyai satu parameter berupa bilangan int?
6. Commit dan push kode program ke Github

1. Dua karakteristik class atau object:

(Penyembunyian): Class menggabungkan data (atribut) dan fungsi (method) menjadi satu kesatuan. Detail implementasi disembunyikan dari pengguna, hanya interface yang terbuka.

(Pewarisan): Class dapat mewarisi sifat dan kemampuan dari class lain, memungkinkan reuse code dan pengembangan class yang lebih kompleks.

2. Atribut class Buku:

- judul (String): Judul buku
- pengarang (String): Nama pengarang
- halaman (int): Jumlah halaman
- stok (int): Sisa stok buku
- harga (int): Harga buku

3. Method class Buku:

- tampilkaninformasi(): Menampilkan informasi buku (judul, pengarang, halaman, stok, harga)
- terjual(int jml): Mengurangi stok buku berdasarkan jumlah yang terjual
- restock(int jml): Menambah stok buku gantiharga
- (int hrg): Mengubah harga buku

4. Modifikasi

```
void terjual(int jml){  
    if (stok > 0 ){  
        stok -= jml;  
    } else {  
        System.out.println(x:"Jumlah barang Tidak mencukupi");  
    }  
}
```

5. Alasan method restock() memiliki parameter bilangan int: Parameter jml pada method restock() digunakan untuk menentukan berapa banyak stok yang ingin ditambahkan. Hal ini memungkinkan penambahan stok dalam jumlah yang bervariasi.

Percobaan 2: Instansiasi Object, serta Mengakses Atribut dan Method

```
1 package jobsheet2;
2
3 public class BukuMain26 {
4     Run | Debug
5     public static void main(String[] args) {
6
7         Buku26 bk1 = new Buku26();
8         bk1.judul = "Today Ends Tomorrow comes";
9         bk1.pengarang = "Denanda Pratiwi";
10        bk1.halaman = 198;
11        bk1.stok = 13;
12        bk1.harga = 71000;
13
14        bk1.tampilInformasi();
15        bk1.terjual(jml:5);
16        bk1.gantiharga(hrg:60000);
17        bk1.tampilInformasi();
18    }
19
20 }
21
22 }
```

1. Pada class **BukuMain**, tunjukkan baris kode program yang digunakan untuk proses instansiasi! Apa nama object yang dihasilkan?

Buku26 bk1 = new Buku26(); , Objek yang dihasilkan adalah bk1

2. Bagaimana cara mengakses atribut dan method dari suatu objek?

Terdapat dua cara untuk mengakses atribut dan method dari suatu objek:

a. Notasi Titik (.) Untuk mengakses atribut, gunakan notasi namaObjek.namaAtribut. Contoh: bk1.judul. Untuk mengakses method, gunakan notasi namaObjek.namaMethod(). Contoh: bk1.tampilInformasi().

b. Getter dan Setter Getter dan setter adalah method khusus yang digunakan untuk mengakses atribut private. Untuk mendapatkan nilai atribut, gunakan method getter. Contoh: bk1.getJudul(). Untuk mengubah nilai atribut, gunakan method setter. Contoh: bk1.setHarga(60000).

3. Mengapa hasil output pemanggilan method **tampilInformasi()** pertama dan kedua berbeda?

Hasil output pemanggilan method `tampilInformasi()` pertama dan kedua berbeda karena : Pada pemanggilan pertama, nilai atribut masih dalam keadaan awal setelah instansiasi. Pada pemanggilan kedua, nilai atribut telah diubah setelah method `terjual()` dan `gantiHarga()` dipanggil.

Percobaan 3: Membuat Konstruktor

```
1 package jobsheet2;
2
3 public class BukuMain26 {
4     Run | Debug
5     public static void main(String[] args) {
6
7         Buku26 bk1 = new Buku26();
8         bk1.judul = "Today Ends Tomorrow comes";
9         bk1.pengarang = "Denanda Pratiwi";
10        bk1.halaman = 198;
11        bk1.stok = 13;
12        bk1.harga = 71000;
13
14        bk1.tampilInformasi();
15        bk1.terjual(jml:5);
16        bk1.gantiharga(hrg:60000);
17        bk1.tampilInformasi();
18
19        Buku26 bk2 = new Buku26(jud:"Self Reward" , pg:"Maheera Ayesha", hal:160,stok:29,har:59000);
20        bk2.terjual(jml:11);
21        bk2.tampilInformasi();
22    }
23
24 }
```

Pertanyaan

1. Pada class **Buku** di Percobaan 3, tunjukkan baris kode program yang digunakan untuk mendeklarasikan konstruktor berparameter!

Konstruktornya adalah:

```
public Buku26(String jud, String pg, int hal, int stok, int har){  
    judul = jud;  
    pengarang = pg;  
    halaman = hal;  
    this.stok = stok;  
    harga = har;  
}
```

2. Perhatikan class **BukuMain**. Apa sebenarnya yang dilakukan pada baris program berikut?

```
Buku bk2 = new Buku(jud:"Self Reward", pg:"Maheera Ayesha", hal:160, stok:29, har:59000);
```

`new Buku26("Self Reward", "Maheera Ayesha", 160, 29, 59000)` merupakan pemanggilan konstruktor berparameter pada class **Buku26**. Lima parameter yang diberikan ("Self Reward", "Maheera Ayesha", 160, 29, 59000) digunakan untuk menginisialisasi atribut judul, pengarang, halaman, stok, dan harga dari objek baru yang dibuat.

3. Hapus konstruktor default pada class **Buku**, kemudian compile dan run program. Bagaimana hasilnya? Jelaskan mengapa hasilnya demikian!

Saat konstruktor default dihapus, compiler Java tidak dapat menemukan konstruktor yang sesuai untuk membuat objek baru dari class **Buku**. Hal ini karena Java secara otomatis menyediakan konstruktor default jika tidak ada konstruktor lain yang didefinisikan. Ketika konstruktor default dihapus, tidak ada konstruktor yang tersedia untuk menginisialisasi objek baru.

4. Setelah melakukan instansiasi object, apakah method di dalam class **Buku** harus diakses secara berurutan? Jelaskan alasannya!

Tidak, method di dalam class **Buku** tidak harus diakses secara berurutan setelah melakukan instansiasi object. Akses terhadap method bersifat independent

5. Buat object baru dengan nama **buku<NamaMahasiswa>** menggunakan konstruktor berparameter dari class **Buku**!

```
Buku26 bkSatria = new Buku26 (jud:"Belajar kuyy", pg:"Satria Wiguna", hal:250, stok:89, har:78000);  
bkSatria.terjual(jml:20);  
bkSatria.tampilInformasi(); //
```

Latihan Praktikum:

1. Pada class Buku yang telah dibuat, tambahkan tiga method yaitu **hitungHargaTotal()**, **hitungDiskon()**, dan **hitungHargaBayar()** dengan penjelasan sebagai berikut:

```
public int hitungHargaTotal(){
    totalharga = harga * terjual;
    return totalharga;
}

public double hitungDiskon() {
    if (totalharga > 150000) {
        hargadiskon = totalharga * 0.12;
    }
    else if(75000 > totalharga && totalharga < 150000){
        hargadiskon = totalharga * 0.10;
    }
    else{
        hargadiskon = 0; // Tidak ada diskon
    }
    return hargadiskon;
}

int hitungHargaBayar(int total, int diskon){
    hargabayar = (int) (totalharga-hargadiskon);
    return hargabayar;
}

}
```

```
Judul : Today Ends Tomorrow comes
Pengarang : Denanda Pratiwi
Jumlah halaman : 198
Sisa stok : 8
Harga : RP 71000
Harga Total 355000
Harga Diskon 42600.0
Harga bayar 312400
Judul : Today Ends Tomorrow comes
Pengarang : Denanda Pratiwi
Jumlah halaman : 198
Sisa stok : 8
Harga : RP 60000
Harga Total 355000
Harga Diskon 42600.0
Harga bayar 312400
Judul : Self Reward
Pengarang : Maheera Ayesha
Jumlah halaman : 160
Sisa stok : 18
Harga : RP 59000
Harga Total 649000
Harga Diskon 77880.0
Harga bayar 571120
Judul : Belajar kuyy
Pengarang : Satria Wiguna
Jumlah halaman : 250
Sisa stok : 89
Harga : RP 78000
Harga Total 78000
Harga Diskon 0.0
Harga bayar 78000
```

2. Buat program berdasarkan class diagram berikut ini! Dragon

```
(1 2)
PS D:\Tugas\Codingan\Semester 2>
```

```
1 public class Dragon {
2     int x , y, width, height;
3     public Dragon (int x1, int y1, int w, int h){
4         x= x1;
5         y= y1;
6         width = w;
7         height = h;
8     }
9
10    void moveLeft(){
11        if (x>0){
12            x = x-1;
13            PrintPosition();
14            detecCollision();
15        }
16    }
17
18    void moveRight(){
19        if (x< width){
20            x = x+1;
21            PrintPosition();
22            detecCollision();
23        }
24    }
25
26    void moveUp(){
27        if (y >0){
28            y = y-1;
29            PrintPosition();
30            detecCollision();
31        }
32    }
33
34    void moveDown(){
35        if (y<height){
36            y = y+1;
37            PrintPosition();
38            detecCollision();
39        }
40    }
41
42    void PrintPosition(){
43        System.out.printf("%d %d\n", x, y);
44    }
45
46    void detecCollision () {
47        if (x<= 0 || y<=0 || x>= width || y >= height){
48            System.out.println("Game Over");
49        }
50    }
51
52    public static void main(String[] args) {
53        Dragon d = new Dragon(1, 2, 0, 0);
54        d.PrintPosition();
55        d.moveRight();
56        d.moveDown();
57    }
58 }
```