

ALGORITMA DAN STRUKTUR DATA

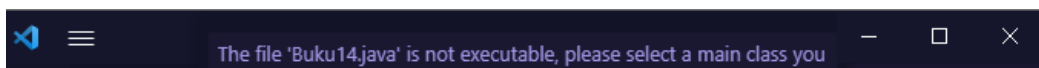
Jobsheet 2 Object

Lavina 2341760062

1. Praktikum 1

```
1  public class Buku14 {  
2  
3      String judul, pengarang;  
4      int halaman, stok, harga;  
5  
6      void tampilInformasi() {  
7          System.out.println("Judul: " + judul);  
8          System.out.println("Pengarang: " + pengarang);  
9          System.out.println("Jumlah halaman: " + halaman);  
10         System.out.println("Sisa stok: " + stok);  
11         System.out.println("Harga: Rp " + harga);  
12     }  
13  
14     void terjual(int jml) {  
15         stok -= jml;  
16     }  
17  
18     void restock(int jml) {  
19         stok += jml;  
20     }  
21  
22     void gantiHarga(int hrg) {  
23         harga = hrg;  
24     }  
25 }
```

Hasil setelah decompile dan run:



Pertanyaan

1. Sebutkan dua karakteristik class atau object!

Jawab: Class tidak memiliki fungsi main dan isinya bisa berupa atribut yang dideklarasikan seperti variabel biasa, dan method.

2. Perhatikan class Buku pada Praktikum 1 tersebut, ada berapa atribut yang dimiliki oleh class Buku? Sebutkan apa saja atributnya!

Jawab: Ada 5 yaitu, judul, pengarang, halaman, stok dan harga.

3. Ada berapa method yang dimiliki oleh class tersebut? Sebutkan apa saja methodnya!

Jawab: Ada 4 yaitu, tampilInformasi, terjual, restock dan gantiHarga

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)!

```
1 void terjual(int jml) {  
2     if (stok > 0) {  
3         stok -= jml;  
4     } else {  
5         System.out.println("Stok Habis");  
6     }  
7 }  
8
```

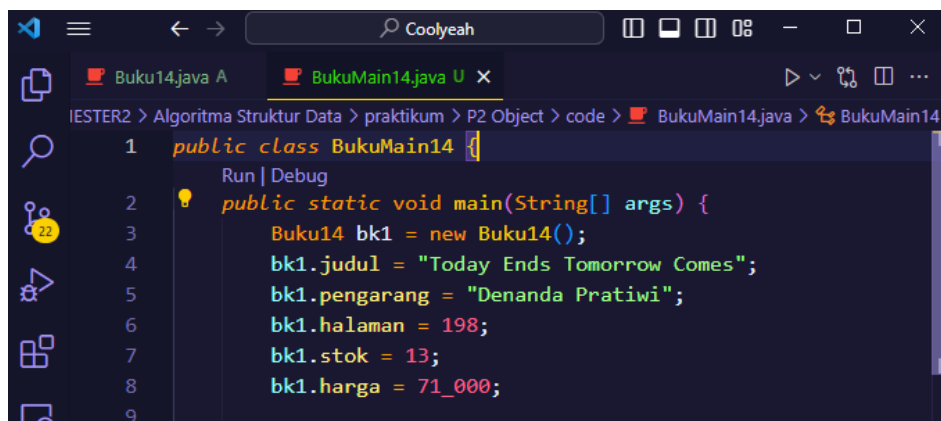
5. Menurut Anda, mengapa method restock() mempunyai satu parameter berupa bilangan int?

Jawab: Gunanya untuk menangkap inputan dari pengguna berupa jumlah stok buku yang ditambahkan.

6. Commit dan push kode program ke Github

Link:

2. Praktikum 2



```
1 public class BukuMain14 {  
2     public static void main(String[] args) {  
3         Buku14 bk1 = new Buku14();  
4         bk1.judul = "Today Ends Tomorrow Comes";  
5         bk1.pengarang = "Denanda Pratiwi";  
6         bk1.halaman = 198;  
7         bk1.stok = 13;  
8         bk1.harga = 71_000;  
9     }  
10 }
```

```
10      bk1.tampilInformasi();
11      bk1.terjual(jml:5);
12      bk1.gantiHarga(hrg:60_000);
13      bk1.tampilInformasi();
14  }
15 }
```

PROBLEMS OUTPUT TERMINAL ...

Run: BukuMain14

```
680d90275@redhat.java\jdt_ws\Coolyeh14a4c95\bin' 'BukuMain14'
Judul: Today Ends Tomorrow Comes
Pengarang: Denanda Pratiwi
Jumlah halaman: 198
Sisa stok: 13
Harga: Rp 71000
Judul: Today Ends Tomorrow Comes
Pengarang: Denanda Pratiwi
Jumlah halaman: 198
Sisa stok: 8
Harga: Rp 60000
PS D:\Coolyeh>
```

Pertanyaan

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

Jawab: Proses instansiasi ada di baris ke 3, nama objek yang dihasilkan adalah bk1

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

Jawab: Setelah melakukan instansiasi dari class menjadi objek, untuk mengakses atribut dan methodnya yaitu dengan menuliskan nama objek lalu diikuti notasi titik ".". " lalu nama atribut atau method yang ingin digunakan. Contohnya: objek.namaAtribut atau objek.namaMethod.

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

Jawab: Karena pada pemanggilan kedua didahului proses dari method terjual yang mengurangi stok buku dan juga method gantiHarga yang mengubah harga buku sehingga jumlah stok dan harganya berubah saat pemanggilan kedua.

3. Praktikum 3

Buku14.java

```
1  public Buku14() {
2
3  }
4
5  public Buku14(String jud, String pg, int hal, int stok, int har) {
6      judul = jud;
7      pengarang = pg;
8      halaman = hal;
9      this.stok = stok;
10     harga = har;
11 }
12
```

BukuMain.java

```
1      Buku14 bk2 = new Buku14("Self Reward", "Maheera Ayesha", 160, 29, 59_000);
2      bk2.terjual(11);
3      bk2.tampilInformasi();
4
```

Hasil setelah dicompile dan run:



```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS
PS D:\Coolyeh> & 'C:\Program Files\Java\jdk-21\bin\java.exe' '-agentlib:jdwp=transport=dt_socket,server=n,susp
DetailsInExceptionMessages' '-cp' 'C:\Users\Lavina\AppData\Roaming\Code\User\workspaceStorage\c46e7221177a4bd3a2
n14'
Judul: Today Ends Tomorrow Comes
Pengarang: Denanda Pratiwi
Jumlah halaman: 198
Sisa stok: 13
Harga: Rp 71000
Judul: Today Ends Tomorrow Comes
Pengarang: Denanda Pratiwi
Jumlah halaman: 198
Sisa stok: 8
Harga: Rp 60000
Judul: Self Reward
Pengarang: Maheera Ayesha
Jumlah halaman: 160
Sisa stok: 18
Harga: Rp 59000
PS D:\Coolyeh>
```

Pertanyaan

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

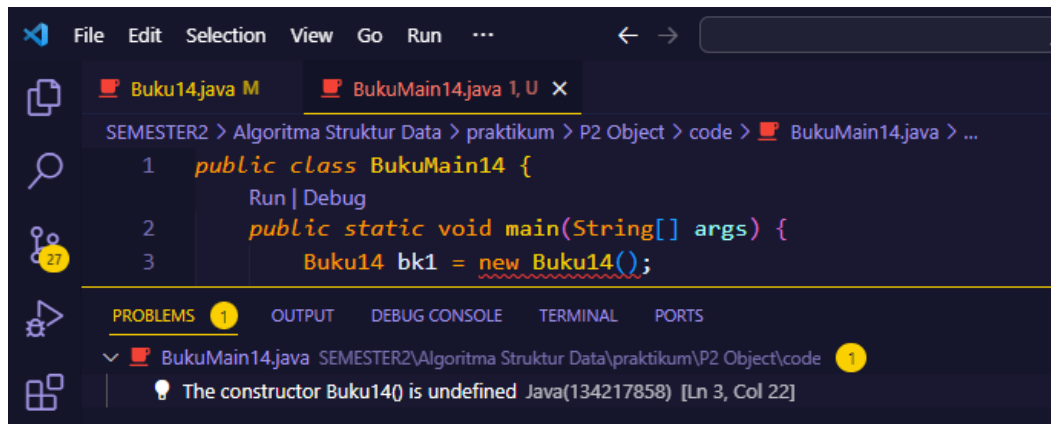
Jawab: Baris kode yang menunjukkan deklarasi kode konstruktor berparameter yaitu baris ke 5 – 11.

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);
```

Jawab: Yang terjadi dibaris program tersebut yaitu instansiasi objek sekaligus pemberian nilai untuk setiap atribut dari class Buku.

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



```
1 public class BukuMain14 {  
2     public static void main(String[] args) {  
3         Buku14 bk1 = new Buku14();  
    }  
}
```

PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS

1 BukuMain14.java SEMESTER2\Algoritma Struktur Data\praktikum\P2 Object\code 1

The constructor Buku14() is undefined Java(134217858) [Ln 3, Col 22]

Terjadi error karena, konstruktor berparameter dibuat agar wajib diisi nilai parameternya saat melakukan instansiasi, konstruktor default dibuat sehingga jika saat instansiasi tidak diisi nilainya komputer masih tetap bisa menjalankan programnya, sehingga saat konstruktor default dihapus maka saat instansiasi harus mengisi nilai dari parameternya atau tidak boleh kosong seperti pada baris

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

Jawab: Tidak harus, mengakses method dari sebuah class bersifat independen, artinya method bisa digunakan dalam urutan apapun.

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

```
1     Buku14 bukuLavina = new Buku14("Harry Potter", "JK Rowling", 250, 20, 99_000);  
2     bukuLavina.terjual(2);  
3     bukuLavina.tampilInformasi();  
4
```

6. Commit dan push kode program ke Github