

Tugas Praktikum Pertemuan 4 - Relasi Kelas



Muhammad Farrel Caesarian
TI-2D/16
2341720053
POLITEKNIK NEGERI MALANG

Percobaan 1

Hasil :

```
● Merk Laptop = Thinkpad  
Merk Processor = Intel I5  
Cache Memory = 3.00  
Merk Laptop = Thinkpad  
Merk Processor = Intel i5  
Cache Memory = 4.00
```

Pertanyaan

Berdasarkan percobaan 1, jawablah pertanyaan-pertanyaan yang terkait:

1. Di dalam *class* Processor dan *class* Laptop , terdapat method *setter* dan *getter* untuk masing- masing atributnya. Apakah gunanya *method setter* dan *getter* tersebut ?

Jawab: setter berfungsi untuk memberikan nilai pada atribut, sedangkan getter berfungsi untuk menampilkan atribut.

2. Di dalam *class* Processor dan *class* Laptop, masing- masing terdapat konstruktor default dan konstruktor berparameter. Bagaimanakah beda penggunaan dari kedua jenis konstruktor tersebut ?

Jawab: Konstruktor default tidak menerima parameter, sehingga kita tidak harus memberikan nilai pada atribut saat membuat objek. Berbeda dengan konstruktor berparameter yang mengharuskan kita untuk menuliskan atribut saat membuat objek.

3. Perhatikan *class* Laptop, di antara 2 atribut yang dimiliki (*merk* dan *proc*), atribut manakah yang bertipe *object* ?

Jawab: Atribut yang bertipe object adalah atribut *proc*, karena *proc* merupakan object dari kelas Processor yang menandakan bahwa *proc* bukanlah tipe data primitif seperti *int* atau *String* melainkan sebuah instance dari kelas lain.

4. Perhatikan *class* Laptop, pada baris manakah yang menunjukkan bahwa *class* Laptop memiliki relasi dengan *class* Processor ?

Jawab:

```
private Processor proc;
```

5. Perhatikan pada *class* Laptop , Apakah guna dari sintaks *proc.info()* ?

Jawab: Untuk menampilkan informasi dari processor.

6. Pada *class* MainPercobaan1, terdapat baris kode:

```
Laptop l = new Laptop("Thinkpad", p);
```

Apakah *p* tersebut ? Dan apakah yang terjadi jika baris kode tersebut diubah menjadi:

```
Laptop l = new Laptop("Thinkpad", new Processor("Intel  
i5", 3));
```

Bagaimanakah hasil program saat dijalankan, apakah ada perubahan ?

Jawab: *p* adalah object dari kelas Processor, jika kode diganti menjadi kode seperti diatas, outputnya sama dengan kode awal.

Percobaan 2

Pertanyaan

Hasil Percobaan:

```
Biaya Total = 1100000
```

1. Perhatikan *class* Pelanggan. Pada baris program manakah yang menunjukkan bahwa *class* Pelanggan memiliki relasi dengan *class* Mobil dan *class* Sopir ?

Jawab:

```
private Mobil mobil;  
private Sopir sopir;
```

2. Perhatikan *method* hitungBiayaSopir pada *class* Sopir, serta *method* hitungBiayaMobil pada *class* Mobil. Mengapa menurut Anda *method* tersebut harus memiliki argument hari ?

Jawab:

Agar perhitungan biaya dapat diproses.

3. Perhatikan kode dari *class* Pelanggan. Untuk apakah perintah mobil.hitungBiayaMobil(hari) dan sopir.hitungBiayaSopir(hari) ?

Jawab:

Untuk menghitung biaya total untuk pelanggan.

4. Perhatikan *class* MainPercobaan2. Untuk apakah sintaks p.setMobil(m) dan p.setSopir(s) ?

Jawab: Pada kelas MainPercobaan2, sintaks p.setMobil(m) dan p.setSopir(s) digunakan untuk menghubungkan objek Mobil dan objek Sopir dengan objek Pelanggan

5. Perhatikan *class* MainPercobaan2. Untuk apakah proses p.hitungBiayaTotal() tersebut ?

Jawab: menghitung biaya total untuk pelanggan.

6. Perhatikan *class* MainPercobaan2, coba tambahkan pada baris terakhir dari *method* main dan amati perubahan saat di- run!Page 6 of 10

```
System.out.println(p.getMobil().getMerk());
```

Jadi untuk apakah sintaks p.getMobil().getMerk() yang ada di dalam *method* main tersebut?

Jawab:

```
Biaya Total = 1100000  
Avanza
```

Menampilkan merk mobil yang disewa pelanggan.

Percobaan 3

Hasil Percobaan

```
Nama: Gaya Baru  
Kelas: Bisnis  
Masinis: Nip: 1234  
Nama: Spongebob Squarepants  
  
Asisten: Nip: 4567  
Nama: Patrick Star
```

Pertanyaan

1. Di dalam *method* info() pada *class* KeretaApi, baris this.masinis.info() dan

this.asisten.info() digunakan untuk apa ?

Jawab: Untuk menampilkan info tentang masinis dan asisten.

2. Buatlah *main* program baru dengan nama *class* MainPertanyaan pada *package* yang sama. Tambahkan kode berikut pada *method* main() ! Pegawai masinis = new

Pegawai("1234", "Spongebob Squarepants"); KeretaApi keretaApi = new

KeretaApi("Gaya Baru", "Bisnis", masinis);

System.out.println(keretaApi.info());

Jawab:

```
public class MainPertanyaan{
    Run | Debug
    public static void main(String[] args) {
        Pegawai masinis = new Pegawai(nip:"1234", nama:"Spongebob Squarepants"); KeretaApi keretaApi =
        new KeretaApi(nama:"Gaya Baru", kelas:"Bisnis", masinis);
        System.out.println(keretaApi.info());
    }
}
```

3. Apa hasil output dari *main* program tersebut ? Mengapa hal tersebut dapat terjadi ?

Jawab:

```
Exception in thread "main" java.lang.NullPointerException: Cannot invoke "Pegawai.info()" because "this.asisten" is null
    at KeretaApi.info(MainPercobaan3.java:73)
    at MainPercobaan3.main(MainPercobaan3.java:80)
```

Hasilnya error karena

4. Perbaiki *class* KeretaApi sehingga program dapat berjalan !

Jawab:

```
public String info() {
    String info = "";
    info += "Nama Kereta: " + this.nama + "\n";
    info += "Kelas: " + this.kelas + "\n";
    info += "Masinis: " + this.masinis.info() + "\n";
    if (this.asisten != null) {
        info += "Asisten: " + this.asisten.info() + "\n";
    } else {
        info += "Asisten: Tidak ada\n";
    }
    return info;
}
```

```
Nama Kereta: Gaya Baru
Kelas: Bisnis
Masinis: Nip: 1234
Nama: Spongebob Squarepants
Asisten: Tidak ada
```

Percobaan 4

Hasil Percobaan

```
Kode: A
Nomor: 1
Penumpang: KTP: 12345
Nama: Mr. Krab
```

```
Nomor: 2
Nomor: 3
Nomor: 4
Nomor: 5
Nomor: 6
Nomor: 7
Nomor: 8
Nomor: 9
Nomor: 10
```

Pertanyaan

1. Pada *main* program dalam *class* MainPercobaan4, berapakah jumlah kursi dalam Gerbong A ?

Jawab: 10

2. Perhatikan potongan kode pada *method* info() dalam *class* Kursi. Apa maksud kode tersebut ?

```
...
if (this.penumpang != null) {
    info += "Penumpang: " + penumpang.info() + "\n";
}
...
```

Jawab: saat ada penumpang, maka informasi penumpang akan ditambahkan.

3. Mengapa pada *method* setPenumpang() dalam *class* Gerbong, nilai nomor dikurangi dengan angka 1 ?

Jawab: Ketika nomor kursi yang diberikan adalah 1 (kursi pertama), kita mengurangi 1 menjadi 0, sehingga kita mengakses elemen pertama dari arrayKursi. Jika nomor kursi yang diberikan adalah 2, kita mengakses elemen kedua, dan seterusnya.

4. Instansiasi objek baru budi dengan tipe Penumpang, kemudian masukkan objek baru tersebut pada gerbong dengan gerbong.setPenumpang(budi, 1). Apakah yang terjadi ?

Jawab:

```
Kode: A
Nomor: 1
Penumpang: KTP: 11111
Nama: Budi
```

Mr. Krab akan terhapus dari list penumpang

5. Modifikasi program sehingga tidak diperkenankan untuk menduduki kursi yang sudah ada penumpang lain !

Jawab:

```

public void setPenumpang(Penumpang penumpang, int nomor) {
    if (this.arrayKursi[nomor - 1].getPenumpang() != null) {
        System.out.println("Kursi nomor " + nomor + " sudah terisi. Silakan pilih kursi lain");
        System.out.println();
    } else {
        this.arrayKursi[nomor - 1].setPenumpang(penumpang);
        System.out.println("Penumpang " + penumpang.getNama() + " berhasil duduk di kursi no
    }
}

```

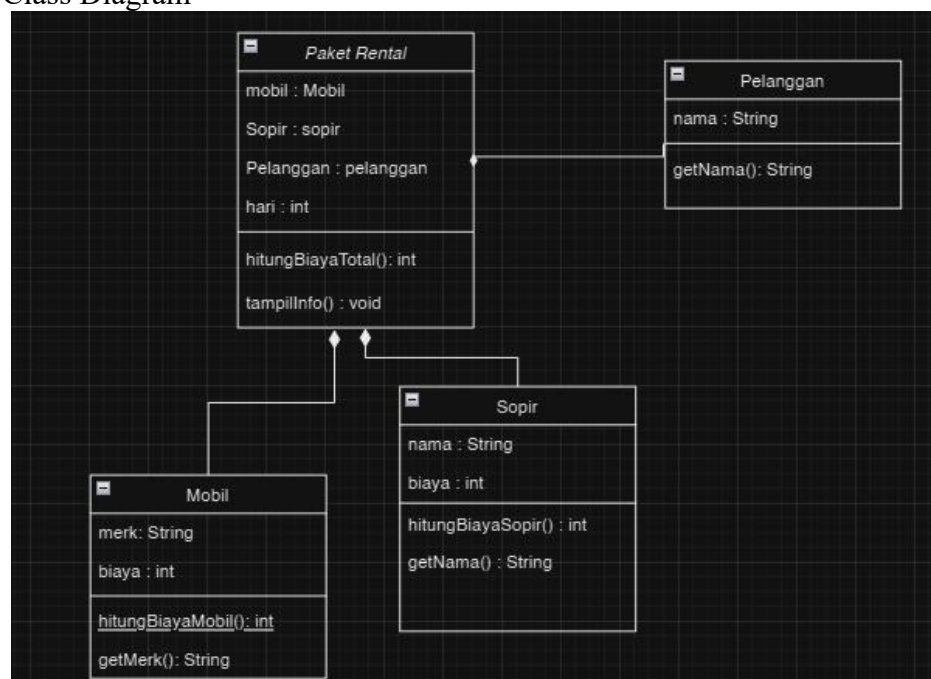
Kursi nomor 1 sudah terisi. Silakan pilih kursi lain.

Kode: A
 Nomor: 1
 Penumpang: KTP: 12345
 Nama: Mr. Krab

Nomor: 2
 Nomor: 3
 Nomor: 4
 Nomor: 5
 Nomor: 6
 Nomor: 7
 Nomor: 8
 Nomor: 9
 Nomor: 10

Tugas

Class Diagram



Nama Pelanggan: Caesarian
 Mobil: Palisade
 Sopir: Farrel
 Lama sewa: 7 hari
 Total Biaya: 3850000


```

1 class Mobil {
2     private String merk;
3     private int biaya;
4
5     public Mobil(String merk, int biaya) {
6         this.merk = merk;
7         this.biaya = biaya;
8     }
9
10    public int hitungBiayaMobil(int hari) {
11        return biaya * hari;
12    }
13
14    public String getMerk() {
15        return merk;
16    }
17 }
18
19 class Sopir {
20     private String nama;
21     private int biaya;
22
23     public Sopir(String nama, int biaya) {
24         this.nama = nama;
25         this.biaya = biaya;
26     }
27
28     public int hitungBiayaSopir(int hari) {
29         return biaya * hari;
30     }
31
32     public String getNama() {
33         return nama;
34     }
35 }
36
37 class Pelanggan {
38     private String nama;
39
40     public Pelanggan(String nama) {
41         this.nama = nama;
42     }
43
44     public String getNama() {
45         return nama;
46     }
47 }
48
49 class PaketRental {
50     private Mobil mobil;
51     private Sopir sopir;
52     private Pelanggan pelanggan;
53     private int hari;
54
55     public PaketRental(Mobil mobil, Sopir sopir, Pelanggan pelanggan, int hari) {
56         this.mobil = mobil;
57         this.sopir = sopir;
58         this.pelanggan = pelanggan;
59         this.hari = hari;
60     }
61
62     public int hitungBiayaTotal() {
63         return mobil.hitungBiayaMobil(hari) + sopir.hitungBiayaSopir(hari);
64     }
65
66     public void tampilInfo() {
67         System.out.println("Nama Pelanggan: " + pelanggan.getNama());
68         System.out.println("Mobil: " + mobil.getMerk());
69         System.out.println("Sopir: " + sopir.getNama());
70         System.out.println("Lama sewa: " + hari + " hari");
71         System.out.println("Total Biaya: " + hitungBiayaTotal());
72     }
73 }
74
75 public class TugasJabsheet4 {
76     public static void main(String[] args) {
77         Mobil mobil = new Mobil("Pallisade", 450000);
78         Sopir sopir = new Sopir("Farrel", 100000);
79         Pelanggan pelanggan = new Pelanggan("Caesarian");
80
81         PaketRental paketRental = new PaketRental(mobil, sopir, pelanggan, 7);
82         paketRental.tampilInfo();
83     }
84 }
85

```

