

**LAPORAN PRAKTIKUM**  
**PEMROGRAMAN BERORIENTASI OBJECT**  
**JOBSHEET 4**



**Sahrul Ramadhani**

**244107020058**

**2H**

**D-IV Teknik Informatika**

**Teknologi Informasi**

**Politeknik Negeri Malang**

**2025**

## Hasil Percobaan 1

```
Merk Laptop = ThinkPad  
Merk Laptop = Intel i5  
Cache Memory = 3.00  
Merk Laptop = ThinkPad  
Merk Laptop = Intel i5  
Cache Memory = 4.00
```

### Jawaban pertanyaan Percobaan 1

1. - Setter digunakan untuk mengubah nilai atribut/memberikan nilai isi sesuai tipedatanya.  
- Getter ini diunakan untuk mengambil isi atribut yang telah di tetapkan oleh si setter tadi
2. Konstruktor default itu parameternya di kosongin dulu, bisa di pakai atau di isi belakangan dengan method setter, sedangkan konstruktor berparameter langsung di berikan nilai saat object dibuat.
3. Proc yang menyimpan nilai dari object Processor
4. Pada kode :

```
public Laptop(String merk, Processor proc) {  
    this.merk = merk;  
    this.proc = proc;  
}
```

Menandakan laptop punya Processor dengan merelasikan keduanya.

5. Menampilkan processor yang di pilih laptop tersebut, jadi tiap melakukan laptop.info juga akan memanggil dan menampilkan processornya
6. Bisa kok , berarti langsung buat object processor baru di dalam konstruktor laptop tanpa membuat baru secara terpisah contoh kode :

```
Laptop L = new Laptop("ThinkPad", new Processor("Intel i5",  
3) );  
L.info();
```

## Percobaan 2

Hasil RUN:

```
[Running] cd "d:\Pemrograman Berbasis Object"  
MainPercobaan2  
Biaya Total = 1100000  
[Done] exited with code=0 in 1.78 seconds
```

### Jawaban Pertanyaan Percobaan 2

1. Relasinya bisa di lihat pada class pelanggan ada variable yang menggunakan tipe data Mobil dan Sopir, lalu pada konstruktor pelanggan juga ada parameter ber tipe Mobil dan Sopir itu adalah relasinya.
2. Ya karena biayanya itu dihitung perhari berdasarkan jumlah hari, dan tarif harian akan di kalikan dengan jumlah hari
3. - mobil.hitungBiayaMobil(hari): menghitung **total biaya sewa mobil** selama berapa hari (biayaMobilPerHari \* hari).  
- sopir.hitungBiayaSopir(hari): menghitung **total biaya sopir** selama berapa hari (biayaSopirPerHari \* hari).
4. Untuk menghubungkan objek Mobil m dan Sopir s ke object Pelanggan p, agar Pelanggan p memiliki object Mobil m dan Sopir s tertentu yang sudah dibuat
5. Untuk menghitung total semuanya, sewa mobil dan sopir, dengan data biaya Sopir dan Mobil x jumlah hari.
6. Mengambil Object mobil Pelanggan p dan menampilkan merknya, namun pada Class Mobil harus diberikan getmobil untuk mengambil nama/merk mobilnya.

### Percobaan 3

Hasil RUN :

```
mrograman Berbasis Object_b3ef0170\bin
Nama : Gaya Baru
Kelas : Bisnis
Masinis : Nip : 1234
Nama : Spongebob Squarepants

Asisten : Nip : 4567
Nama : Patrick Star

PS D:\Pemrograman Berbasis Object>
```

### Jawaban Pertanyaan Percobaan 3

1. Keduanya untuk memanggil method info yang telah dibuat pada kedua class, yaitu class pegawai dan keretaApi, tujuannya adalah untuk menampilkan info yang ada di dalam masing masing method tsbt.
2. Sudah aku modifikasi
3. Outputnya eror karena argumen yang ada di dalam konstruktor tidak sesuai dengan jumlah wadah konstrutornya yang berjumlah 4, hanya di masukan 3.
4. Ya tinggal tmbahkan sebuah konstruktor lagi yang menerima 3 argumen tanpa asisten dan menambahkan ternary pada keretaApi.infonya

```
public KeretaApi (String nama, String kelas, Pegawai masinis){
    this.nama = nama;
    this.kelas = kelas;
    this.masinis = masinis;
}
```

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

```
Nama : Gaya Baru
Kelas : Bisnis
Masinis : Nip : 1234
Nama : Spongebob Squarepants

Asisten:
-
```

## Percobaan 4

Hasil RUN:

```
70\bin' 'MainPercobaan4'
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
```

### Jawaban Pertanyaan Percobaan 4

1. Jumlah kursi di masukan saat inisiasi new Gerbong tadi sebanyak 10 jumlah Kursi
2. Menggunakan ternary agar hanya menampilkan info jika kursi = terisi agar mencegah nullpointer
3. Ya karena nomor kan pada awal di inisiasikan = 1, jadi ketika akan di isikan ke index harus nomor -1 agar menempati index ke 0 begitu juga nomor dan index selanjutnya.
4. Ya yang terjadi budi akan menempari kursi no1, isi sebelumnya di tulisulang (overwrite), karena program tidak ada pengecekan kursi yang sudah terisi tidak boleh di timpa
5. Disini saya akan modify pada method setpenumpang yang ada diclass gerbong dan memasukan data ke main pada kursi yang sama

```
public void setPenumpang(Penumpang penumpang, int nomor) {
    int idx = nomor - 1
    if (arrayKursi[idx].getPenumpang() != null) {
        System.out.println("Kursi " + nomor + " sudah terisi. Silakan
pilih kursi lain.");
        Return;
    }
    arrayKursi[idx].setPenumpang(penumpang);
}
```

Pada Main :

```
public class MainPercobaan4 {
    public static void main(String[] args) {
        Penumpang p = new Penumpang("12345", "Mr. Krab");
        Gerbong gerbong = new Gerbong("A", 10);
        gerbong.setPenumpang(p, 1);
        System.out.println(gerbong.info());
        Penumpang p1 = new Penumpang("4628", "Rama");
        gerbong.setPenumpang(p1, 1);
        System.out.println(gerbong.info());
    }
}
```

Output :

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

Kursi 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