

## Percobaan

### 3.3 PERTANYAAN

1. Pada class TestMobil, saat kita menambah kecepatan untuk pertama kalinya, mengapa muncul peringatan “Kecepatan tidak bisa bertambah karena Mesin Off!”?

**Jawab :** karena pada saat motor dalam keadaan off, maka tidak bisa ditambah kecepatan. Atau dalam program yang pertama kali dipanggil adalah motor.printStatus(), yang dimana jika kontakOn maka nilainya true, tetapi disitu tidak ada panggilan true(di motor.printStatus()).

2. Mengapat atribut kecepatan dan kontakOn diset private?

**Jawab :** yaitu untuk menyembunyikan atribut internal dari pengguna/ class lain

3. Ubah class Motor sehingga kecepatan maksimalnya adalah 100!

**Jawab :**

```
23 public void tambahKecepatan() {  
24     if (kontakOn == true) {  
25         kecepatan += 5;  
26  
27         if (kecepatan > 100) {  
28             kecepatan = 100;  
29         }  
30     }  
31     else {  
32         System.out.println("Kecepatan tidak bisa bertambah karena mesin Off! \n");  
33     }  
34 }  
35 }
```

### 3.6 Pertanyaan – Percobaan 3 dan 4

1. Apa yang dimaksud getter dan setter?

**Jawab :** Sejenis method untuk mendapat & mengeset value pada atribut. Jika Getter diunakan untuk mendapat value dari atribut dan punya return value, sedangkan setter untuk mengeset value dari atribut dan tidak punya return value.

2. Apa kegunaan dari method getSimpanan()?

**Jawab :** Untuk menampilkan semua jumlah yang disimpan, setelah ada pengurangan atau penambahan data.

3. Method apa yang digunakan untk menambah saldo?

**Jawab :** Method setor

4. Apa yand dimaksud konstruktor?

**Jawab :** Sejenis method tetapi tidak punya return value, nama konstruktor sama dengan nama classnya, dan deksekusi otomatis saat instansiasi

**Nama : Yulia Eka Ardhani**  
**Kelas : TI 2C**  
**Jobsheet 3 Praktikum PBO**

5. Sebutkan aturan dalam membuat konstruktor?

**Jawab :** Nama konstruktor harus sama dengan nama class, konstruktor tidak punya data return.

6. Apakah boleh konstruktor bertipe private?

**Jawab :** Boleh, karena di java konstruktor dapat digunakan di modifier private, protected, public/default.

7. Kapan menggunakan parameter dengan passing parameter?

**Jawab :** Passing Parameter By Value adalah merupakan metode melewati parameter berdasarkan nilainya. Karena, setiap internal java akan melakukan hal tersebut secara otomatis. Cara yang diambil java untuk melewati parameter tergantung dari tipe argument yang digunakan pada saat pemanggilan method. Apabila parameter ber tipe sederhana (seperti int, char, boolean, dll), maka java akan menggunakan metode Passing Parameter By Value. Namun apabila parameter berupa objek, maka java akan menggunakan metode Passing Parameter By Reference

8. Apa perbedaan atribut class dan instansiasi atribut?

**Jawab :** atribut class digunakan untuk atribut yang nilainya antara objek satu dengan dengan objek yang lain dari class yang sama akan bernilai sama, jika instansiasi atribut digunakan untuk atribut yang memiliki nilai yang berbeda-beda antara objek yang satu dengan objek yang lain dari class yang sama

Contohnya seperti atribut String nim diinstansiasi atribut menjadi `nim = ni;`

9. Apa perbedaan class method dan instansiasi method?

**Jawab :** class method adalah perilaku yang dialami objek yang berasal dari class, jika instansiasi method merupakan instansiasi yang digunakan sebagai perilaku atribut atau bukan dari atribut, dan punya atau tidak parameter. ( contohnya melakukan instansiasi method `setJudul` pada class buku dengan parameter yang berisikan ("Fiksi")

## **TUGAS**

2. Pada program diatas, pada class EncapTest kita mengeset age dengan nilai 35, namun pada saat ditampilkan ke layar nilainya 30, jelaskan mengapa.

**Jawab :** Karena pada method `setAge` jika umur lebih dari 30 maka age nya akan diset menjadi 30. Karena yang diinputkan umurnya 35 dan  $35 > 30$  maka yang ditampilkan adalah nilai 30.

3. Ubah program diatas agar atribut age dapat diberi nilai maksimal 30 dan minimal 18.

**Jawab :**

Nama : Yulia Eka Ardhani  
Kelas : TI 2C  
Jobsheet 3 Praktikum PBO

```
28 public void setAge(int newAge) {  
29     if(newAge > 30){  
30         age =30;  
31     }  
32     if(newAge < 18){  
33         age = 18;  
34     }  
35     else{  
36         age = newAge;  
37     }  
38 }
```

```
17 public static void main(String[] args) {  
18     // TODO code application logic here  
19     EncapDemo encap = new EncapDemo();  
20     encap.setName("James");  
21     encap.setAge(35);  
22  
23     EncapDemo encap1 = new EncapDemo();  
24     encap1.setName("Tina");  
25     encap1.setAge(17);  
26  
27     System.out.println("Name : " + encap.getName());  
28     System.out.println("Age : " + encap.getAge());  
29     System.out.println("Name : " + encap1.getName());  
30     System.out.println("Age : " + encap1.getAge());  
31 }
```

Output - Jobsheet3\_PBO (run) ×

```
run:  
Name : James  
Age : 35  
Name : Tina  
Age : 18  
BUILD SUCCESSFUL (total time: 2 seconds)
```

4-6 ( di file java)