



NAMA : Jud Amal Mukhtar
NIM : 2041720168
KELAS : 2C / TI
MATERI : Enkapsulasi

3.3 Pertanyaan

Pertanyaan :

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

Jawab :

karena kontakOn bernilai false, oleh karena itu method tambahKecepatan() mengeluarkan statement itu.

2. Mengapa atribut kecepatan dan kontakOn diset private?

Jawab :

Karena untuk Enkapsulasi / menjadikan informasi yang susah dipakai oleh pengguna menjadi mudah dengan information hiding dan mengaksesnya dengan menggunakan method yang disediakan.

3. Ubah class Motor sehingga kecepatan maksimalnya adalah 100!

Jawab :

Class Motor =

```
22 public void tambahKecepatan(){
23     if(kontakOn==true){
24         kecepatan+=40;
25         if(kecepatan>100){
26             System.out.println("Kecepatan maksimal adalah 100 !");
27             kecepatan=100;
28         }
29     }else{
30         System.out.println("Kecepatan tidak bisa bertambah karena Mesin Off! \n");
31     }
32 }
```



NAMA : Jud Amal Mukhtar
NIM : 2041720168
KELAS : 2C / TI
MATERI : Enkapsulasi

Output =

```
run:
Kontak OFF
Kecepatan 0

Kecepatan tidak bisa bertambah karena Mesin Off!

Kontak ON
Kecepatan 0

Kontak ON
Kecepatan 40

Kontak ON
Kecepatan 80

Kecepatan maksimal adalah 100 !
Kontak ON
Kecepatan 100

Kontak OFF
Kecepatan 0

BUILD SUCCESSFUL (total time: 2 seconds)
```

3.6 Pertanyaan – Percobaan 3 dan 4

Pertanyaan :

1. Apa yang dimaksud getter dan setter?

Jawab :

Getter adalah method untuk mengambil value dari atribut sebuah class yang atributnya dibuat private.

Setter adalah method untuk mengubah value dari atribut sebuah class yang atributnya dibuat private.

2. Apa kegunaan dari method getSimpanan()?

Jawab :

Untuk mengambil value dari atribut *simpanan*.

3. Method apa yang digunakan untuk menambah saldo?

Jawab :

Method setor().

4. Apa yang dimaksud konstruktor?

Jawab :

Sebuah method yang mempunyai nama sama dengan class-nya dan digunakan untuk mengubah value dari atribut di dalam objek ketika pertama kali menginstansiasikannya / default value.



NAMA : Jud Amal Mukhtar
NIM : 2041720168
KELAS : 2C / TI
MATERI : Enkapsulasi

5. Sebutkan aturan dalam membuat konstruktor?

Jawab :

- Nama konstruktor harus sama dengan nama class.
- Tidak boleh diberi access modifier.
- Diberi parameter ketika atribut memerlukan nilai yang spesifik.

6. Apakah boleh konstruktor bertipe private?

Jawab :

tidak boleh.

7. Kapan menggunakan parameter dengan passing parameter?

Jawab :

ketika atribut memerlukan nilai yang spesifik.

8. Apa perbedaan atribut class dan instansiasi atribut?

Jawab :

Atribut class adalah atribut yang berada dalam class-nya sendiri sedangkan instansiasi atribut adalah atribut yang diakses dalam class yang lain.

9. Apa perbedaan class method dan instansiasi method?

Jawab :

class method adalah method yang berada dalam class-nya sendiri sedangkan instansiasi method adalah method yang diakses dalam class yang lain.



NAMA : Jud Amal Mukhtar
NIM : 2041720168
KELAS : 2C / TI
MATERI : Enkapsulasi

5.1 Jawaban:

```
run:
Name : James
Age : 30
BUILD SUCCESSFUL (total time: 2 seconds)
```

5.2 Jawaban :

karena ketika masuk ke method setAge, semua angka yang melebihi 30 akan di-set dengan angka 30 dan angka dibawah 30 akan di-set tanpa diganti nilainya

5.3 Jawaban :

Class EncapDemo =

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

Class EncapTest =

```
17 public static void main(String[] args) {
18     EncapDemo encap=new EncapDemo();
19     encap.setName("James");
20     encap.setAge(35);
21     System.out.println("Name : "+encap.getName());
22     System.out.println("Age : "+encap.getAge());
23     encap.setName("Sam");
24     encap.setAge(15);
25     System.out.println("Name : "+encap.getName());
26     System.out.println("Age : "+encap.getAge());
27 }
```

Output =

```
run:
Name : James
Age : 30
Name : Sam
Age : 18
BUILD SUCCESSFUL (total time: 1 second)
```



NAMA : Jud Amal Mukhtar
NIM : 2041720168
KELAS : 2C / TI
MATERI : Enkapsulasi

5.4 Jawaban:

```
1  ...5 lines
6  package koperasi_10;
7
8  /**...4 lines */
12 public class Anggota {
13     private String Nama,nomorKTP;
14     private int limitPinjaman,jumlahPinjaman;
15     Anggota(String nomorKTP,String Nama, int limitPinjaman){
16         this.nomorKTP=nomorKTP;
17         this.Nama=Nama;
18         this.limitPinjaman=limitPinjaman;
19     }
20
21     public String getNama() {
22         return Nama;
23     }
24     public String getNomorKTP() {
25         return nomorKTP;
26     }
27     public int getLimitPinjaman() {
28         return limitPinjaman;
29     }
30     public int pinjam(int jumlahPinjam){
31         jumlahPinjaman+=jumlahPinjam;
32         if(jumlahPinjaman>getLimitPinjaman()){
33             System.out.println("Maaf, jumlah pinjaman melebihi limit!");
34             jumlahPinjaman-=jumlahPinjam;
35         }
36         return jumlahPinjaman;
37     }
38
39     public int angsur(int jumlahAngsur){
40         jumlahPinjaman-=jumlahAngsur;
41         return jumlahPinjaman;
42     }
43     public int getJumlahPinjaman() {
44         return jumlahPinjaman;
45     }
46 }
```

5.5 Jawaban :

Class Main =

```
27     System.out.println("\nMembayar angsuran 300.000");
28     donny.angsur(300000);
29     System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());
```

Class Anggota =

```
38     public int angsur(int jumlahAngsur){
39         jumlahPinjaman-=jumlahAngsur;
40         if(jumlahAngsur<(0.1*jumlahPinjaman)){
41             System.out.println("Maaf, angsuran harus 10% dari jumlah pinjaman");
42             jumlahPinjaman+=jumlahAngsur;
43         }
44         return jumlahPinjaman;
45     }
```



NAMA : Jud Amal Mukhtar
NIM : 2041720168
KELAS : 2C / TI
MATERI : Enkapsulasi

Output =

```
run:
Nama Anggota: Donny
Limit Pinjaman: 5000000

Meminjam uang 10.000.000...
Maaf, jumlah pinjaman melebihi limit!
Jumlah pinjaman saat ini: 0

Meminjam uang 4.000.000...
Jumlah pinjaman saat ini: 4000000

Membayar angsuran 300.000
Maaf, angsuran harus 10% dari jumlah pinjaman
Jumlah pinjaman saat ini: 4000000

Membayar angsuran 3.000.000
Jumlah pinjaman saat ini: 1000000
BUILD SUCCESSFUL (total time: 1 second)
```

5.6 Jawaban :

Class TestKoperasi =

```
17 public static void main(String[] args) {
18     Scanner jud=new Scanner(System.in);
19     Anggota donny = new Anggota("111333444", "Donny", 5000000);
20     System.out.println("Nama Anggota: " + donny.getNama());
21     System.out.println("Limit Pinjaman: " + donny.getLimitPinjaman());
22     System.out.println("\nMeminjam uang 10.000.000...");
23     donny.pinjam(jud.nextInt());
24     System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());
25     System.out.println("\nMeminjam uang 4.000.000...");
26     donny.pinjam(jud.nextInt());
27     System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());
28     System.out.print("\nMembayar angsuran ");
29     donny.angsur(jud.nextInt());
30     System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());
31     System.out.print("\nMembayar angsuran |");
32     donny.angsur(jud.nextInt());
33     System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());
34 }
```



NAMA : Jud Amal Mukhtar
NIM : 2041720168
KELAS : 2C / TI
MATERI : Enkapsulasi

Output =

```
run:
Nama Anggota: Donny
Limit Pinjaman: 5000000

Meminjam uang 10.000.000...
10000000
Maaf, jumlah pinjaman melebihi limit!
Jumlah pinjaman saat ini: 0

Meminjam uang 4.000.000...
4000000
Jumlah pinjaman saat ini: 4000000

Membayar angsuran 350000
Maaf, angsuran harus 10% dari jumlah pinjaman
Jumlah pinjaman saat ini: 4000000

Membayar angsuran 2000000
Jumlah pinjaman saat ini: 2000000
BUILD SUCCESSFUL (total time: 46 seconds)
```