

NAMA : MILA YUNITA  
KELAS/NOMOR : TI-2C/12  
NIM : 2041720027  
MATA KULIAH : PEMROGRAMAN BERBASIS OBJEK

## JOBSHEET 3

### ENKAPSULASI PBO

#### Pertanyaan – Percobaan 2

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

**Jawab:**

Peringatan tersebut muncul karena untuk memanggil method tambahKecepatan(), value dari atribut kontakOn harus bernilai true dimana hal itu menandakan bahwa motor dalam keadaan hidup. Sedangkan value dari kontakOn masih bernilai default false saat program pertama kali dijalankan yang mana hal tersebut berarti value motor dalam keadaan mati, maka kondisi pada else-lah yang dijalankan. Kondisi pada else tersebut memunculkan output peringatan “Kecepatan tidak bisa bertambah karena Mesin Off!”.

2. Mengapad atribut kecepatan dan kontakOn diset private?

**Jawab:**

Hal tersebut dilakukan agar class motor dapat digunakan dengan baik dan untuk menyembunyikan atribut internal dari pengguna. Dengan atribut kecepatan dan kontakOn diset menjadi private maka program dapat lebih terkontrol dan pengguna hanya perlu mengakses method-method khusus untuk mengakses atribut tersebut. Sehingga pengguna dapat dipermudah dalam menjalankan program.

3. Ubah class Motor sehingga kecepatan maksimalnya adalah 100!

**Jawab:**

```
25 public void tambahKecepatan(){
26     if(kontakOn == true && kecepatan<100){
27         kecepatan += 5;
28     }
29     else{
30         if(kontakOn==false){
31             System.out.println("Kecepatan tidak bisa bertambah karena Mesin Off! \n");
32         }else if(kecepatan==100){
33             System.out.println("Kecepatan tidak bisa ditambahkan karena batas maksimal kecepatan = 100");
34         }
35     }
36 }
```

NAMA : MILA YUNITA  
KELAS/NOMOR : TI-2C/12  
NIM : 2041720027  
MATA KULIAH : PEMROGRAMAN BERBASIS OBJEK

### **Pertanyaan – Percobaan 3 dan 4**

1. Apa yang dimaksud getter dan setter?

**Jawab:**

Getter dan setter adalah sejenis method yang digunakan untuk mendapatkan dan mengeset value dari sebuah atribut.

- Getter digunakan untuk mendapatkan value dari atribut private dan harus memiliki nilai kembalian(return value).
- Setter digunakan untuk mengeset value dari atribut private dan tidak memiliki nilai kembalian.

2. Apa kegunaan dari method getSimpanan()?

**Jawab:**

Method getSimpanan() digunakan untuk mendapatkan value dari atribut simpanan yang bermodifier private, dengan menggunakan return value.

3. Method apa yang digunakan untuk menambah saldo?

**Jawab:**

Method yang digunakan untuk menambah saldo adalah method setor() yang di dalamnya terdapat parameter uang yang bertipe data float yang berfungsi untuk menyimpan nilai jumlah uang yang akan ditambahkan.

4. Apa yang dimaksud konstruktor?

**Jawab:**

Konstruktor adalah suatu method istimewa dari sebuah class yang memiliki ciri-ciri khusus yaitu mempunyai nama method yang sama seperti nama class, tidak memiliki return value, dan dieksekusi saat instansiasi.

5. Sebutkan aturan dalam membuat konstruktor?

**Jawab:**

Aturan dalam membuat konstruktor antara lain:

- Nama konstruktor harus sama seperti nama class
- Dapat menggunakan tambahan parameter
- Tidak memiliki tipe data class
- Tidak memiliki tipe data return
- Hanya dapat dipanggil saat proses instansiasi

NAMA : MILA YUNITA  
KELAS/NOMOR : TI-2C/12  
NIM : 2041720027  
MATA KULIAH : PEMROGRAMAN BERBASIS OBJEK

- Konstruktor tidak boleh menggunakan modifier abstract, static, final, dan synchronized

6. Apakah boleh konstruktor bertipe private?

**Jawab:**

Ya boleh. Dalam pemrograman java, konstruktor dapat memiliki tipe modifier private, protected, public atau default.

7. Kapan menggunakan parameter dengan passing parameter?

**Jawab:**

Penggunaan parameter dengan passing grade dilakukan ketika dibutuhkan nilai yang spesifik pada parameter konstruktor saat instansiasi objek dilakukan.

8. Apa perbedaan atribut class dan instansiasi atribut?

**Jawab:**

Atribut merupakan nilai data yang terdapat pada suatu object yang berasal dari class. Dengan kata lain atribut merupakan karakteristik atau ciri-ciri yang dimiliki objek. Sedangkan instansiasi atribut adalah pembuatan atribut suatu objek dari sebuah class.

9. Apa perbedaan class method dan instansiasi method?

**Jawab:**

Method adalah perilaku atau behaviour yang dapat dilakukan oleh suatu class. Sedangkan instansiasi method adalah pembuatan method suatu objek dari sebuah class.

NAMA : MILA YUNITA  
KELAS/NOMOR : TI-2C/12  
NIM : 2041720027  
MATA KULIAH : PEMROGRAMAN BERBASIS OBJEK

## TUGAS

1. Cobalah program dibawah ini dan tuliskan hasil outputnya

**Jawab:**

### Class EncapDemo

```
6      package tugasJS3_milay;
7
8      /**
9       *
10     * @author VivoBook
11     */
12     public class EncapDemo {
13         private String name;
14         private int age;
15
16         public String getName() {
17             return name;
18         }
19
20         public void setName(String newName) {
21             name = newName;
22         }
23
24         public int getAge() {
25             return age;
26         }
27
28         public void setAge(int newAge) {
29             if(newAge > 30){
30                 age=30;
31             }else{
32                 age = newAge;
33             }
34         }
35
36     }
```

NAMA : MILA YUNITA  
KELAS/NOMOR : TI-2C/12  
NIM : 2041720027  
MATA KULIAH : PEMROGRAMAN BERBASISI OBJEK

### Class EncapTest

```
6 package tugasJS3_milay;
7
8 /**
9  *
10  * @author VivoBook
11  */
12 public class EncapTest {
13     public static void main(String[] args) {
14         EncapDemo encap = new EncapDemo();
15         encap.setName("James");
16         encap.setAge(35);
17
18         System.out.println("Name : " + encap.getName());
19         System.out.println("Age : "+encap.getAge());
20     }
21 }
22
```

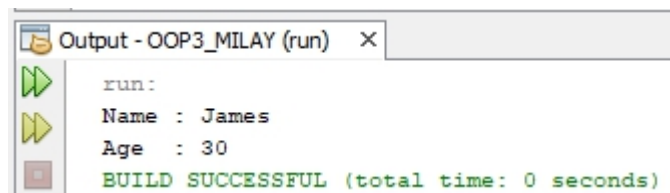
### Hasil Output

Output yang dihasilkan adalah muncul tampilan

Name : James

Age : 30

Seperti pada hasil *capture* di bawah ini.



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

### Jawab:

Hal tersebut karena dalam method `setAge(int newAge)` yang ada di class `EncapDemo` terdapat suatu kondisi yang mana apabila nilai dari parameter `newAge` lebih dari 30 maka atribut `age` akan diset menjadi 30. Dalam class `EncapTest` nilai `age` yang dimasukkan dalam parameter adalah 35, karena 35 lebih besar dari 30 nilai `age` diberikan value 30, dan yang tampil ke layar nilainya juga 30.

NAMA : MILA YUNITA  
KELAS/NOMOR : TI-2C/12  
NIM : 2041720027  
MATA KULIAH : PEMROGRAMAN BERBASIS OBJEK

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

**Jawab:**

```
28 public void setAge(int newAge) {  
29  
30     if (newAge > 30){  
31         System.out.println("Umur " + newAge+ " melebihi batas umur maksimal(max = 30)");  
32         age = 30;  
33     }  
34     else if(newAge < 18){  
35         System.out.println("Umur " + newAge+ " kurang dari batas umur minimal(min = 18)");  
36         age = 18;  
37     }  
38     else{  
39         age = newAge;  
40     }  
41  
42     /*if(newAge > 30){  
43         age=30;  
44     }else{  
45         age = newAge;  
46     }*/  
47 }
```

NAMA : MILA YUNITA  
KELAS/NOMOR : TI-2C/12  
NIM : 2041720027  
MATA KULIAH : PEMROGRAMAN BERBASISI OBJEK

4. Pada sebuah sistem informasi koperasi simpan pinjam, terdapat class Anggota yang memiliki atribut antara lain nomor KTP, nama, limit peminjaman, dan jumlah pinjaman. Anggota dapat meminjam uang dengan batas limit peminjaman yang ditentukan. Anggota juga dapat mengangsur pinjaman. Ketika Anggota tersebut mengangsur pinjaman, maka jumlah pinjaman akan berkurang sesuai dengan nominal yang diangsur. Buatlah class Anggota tersebut, berikan atribut, method dan konstruktor sesuai dengan kebutuhan. Uji dengan TestKoperasi berikut ini untuk memeriksa apakah class Anggota yang anda buat telah sesuai dengan yang diharapkan.

**Jawab:**

### **Class Anggota**

```
6 package tugasJS3_milay;
7
8 /**
9  *
10  * @author VivoBook
11  */
12 public class Anggota {
13     private String ktp;
14     private String nama;
15     private int limitPinjaman=0;
16     private int jumlahPinjaman;
17
18     public Anggota(String ktp, String nama, int limitPinjaman) {
19         this.ktp = ktp;
20         this.nama = nama;
21         this.limitPinjaman = limitPinjaman;
22         jumlahPinjaman=0;
23     }
24
25     public String getKtp() {
26         return ktp;
27     }
28
29     public String getNama() {
30         return nama;
31     }
32
33     public int getLimitPinjaman() {
34         return limitPinjaman;
35     }
36 }
```

NAMA : MILA YUNITA  
KELAS/NOMOR : TI-2C/12  
NIM : 2041720027  
MATA KULIAH : PEMROGRAMAN BERBASISI OBJEK

```
37 public int getJumlahPinjaman() {  
38     return jumlahPinjaman;  
39 }  
40  
41 public void pinjam(int newPinjam){  
42     int jumlah= jumlahPinjaman + newPinjam;  
43  
44     if(jumlah <= limitPinjaman){  
45         jumlahPinjaman += newPinjam;  
46     }else{  
47         System.out.println("Maaf, jumlah pinjaman melebihi limit.");  
48     }  
49 }  
50  
51 public void angsur(int newAngsur){  
52  
53     if(jumlahPinjaman>0 && newAngsur<=jumlahPinjaman){  
54         jumlahPinjaman -= newAngsur;  
55     }  
56     else if(newAngsur>jumlahPinjaman && jumlahPinjaman>0){  
57         int jumlah=jumlahPinjaman;  
58         jumlahPinjaman=0;  
59  
60         System.out.println("Uang Kembalian: "+(newAngsur-jumlah));  
61     }  
62     else{  
63         System.out.println("Tidak ada pinjaman");  
64     }  
65 }  
  
66  
67 public void tampilInfo(){  
68     System.out.println("Nomor KTP : "+ktp);  
69     System.out.println("Nama : " + nama);  
70     System.out.println("Limit Pinjaman : "+limitPinjaman);  
71     System.out.println("Jumlah Pinjaman : "+jumlahPinjaman);  
72 }  
73 }
```

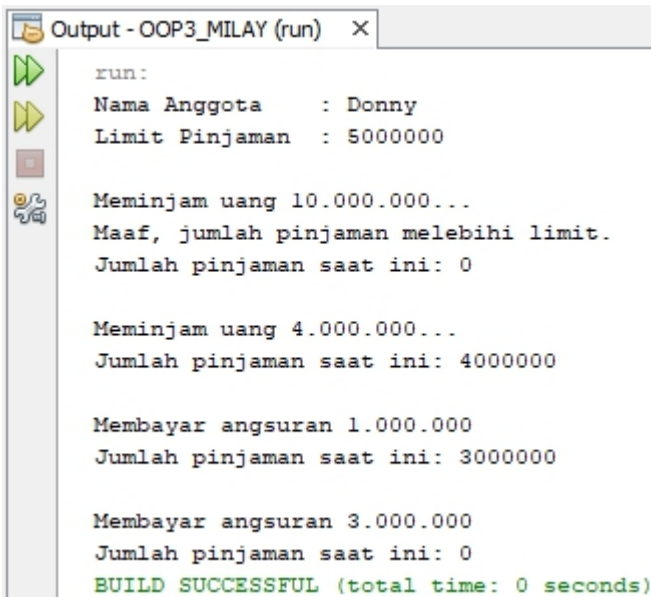


NAMA : MILA YUNITA  
KELAS/NOMOR : TI-2C/12  
NIM : 2041720027  
MATA KULIAH : PEMROGRAMAN BERBASISI OBJEK

### Class TestKoperasi

```
6 package tugasJS3_milay;
7
8 /**
9  *
10  * @author VivoBook
11  */
12 public class TestKoperasi {
13     public static void main(String[] args) {
14         Anggota donny = new Anggota("111333444", "Donny", 5000000);
15
16         System.out.println("Nama Anggota : " + donny.getNama());
17         System.out.println("Limit Pinjaman : " + donny.getLimitPinjaman());
18
19         System.out.println("\nMeminjam uang 10.000.000...");
20         donny.pinjam(10000000);
21         System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());
22
23         System.out.println("\nMeminjam uang 4.000.000...");
24         donny.pinjam(4000000);
25         System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());
26
27         System.out.println("\nMembayar angsuran 1.000.000");
28         donny.angsur(1000000);
29         System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());
30
31         System.out.println("\nMembayar angsuran 3.000.000");
32         donny.angsur(3000000);
33         System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());
```

### 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 1.000.000
Jumlah pinjaman saat ini: 3000000

Membayar angsuran 3.000.000
Jumlah pinjaman saat ini: 0
BUILD SUCCESSFUL (total time: 0 seconds)
```

NAMA : MILA YUNITA  
KELAS/NOMOR : TI-2C/12  
NIM : 2041720027  
MATA KULIAH : PEMROGRAMAN BERBASISI OBJEK

5. Modifikasi soal no. 4 agar nominal yang dapat diangsur minimal adalah 10% dari jumlah pinjaman saat ini. Jika mengangsur kurang dari itu, maka muncul peringatan "Maaf, angsuran harus 10% dari jumlah pinjaman"

**Jawab:**

```
51 public void angsur(int newAngsur){
52
53     if(jumlahPinjaman>0 && newAngsur<=jumlahPinjaman && newAngsur >= hitungAngsurMinimal()){
54         jumlahPinjaman -= newAngsur;
55     }
56     else if(newAngsur>jumlahPinjaman && jumlahPinjaman>0 && newAngsur >= hitungAngsurMinimal()){
57         int jumlah=jumlahPinjaman;
58         jumlahPinjaman=0;
59         System.out.println("Uang Kembalian: "+(newAngsur-jumlah));
60     }
61     else if(newAngsur < hitungAngsurMinimal()){
62         System.out.println("Maaf, angsuran harus 10% dari jumlah pinjaman");
63     }
64     else{
65         System.out.println("Maaf, angsuran tidak dilakukan. Tidak ada pinjaman");
66     }
67 }
68
69 private int hitungAngsurMinimal(){
70     return 10*jumlahPinjaman/100;
71 }
```

6. Modifikasi class TestKoperasi, agar jumlah pinjaman dan angsuran dapat menerima input dari console.

**Jawab:**

```
6 package tugasJS3_milay;
7
8 import java.util.Scanner;
9
10 /**
11  *
12  * @author VivoBook
13  */
14 public class TestKoperasi {
15     public static void main(String[] args) {
16         Scanner scMila = new Scanner(System.in);
17
18         Anggota donny = new Anggota("111333444", "Donny", 5000000);
19
20         do{
21             System.out.println("-----");
22             System.out.println("Nama Anggota : " + donny.getNama());
23             System.out.println("Limit Pinjaman : " + donny.getLimitPinjaman());
24             System.out.println("=====MENU=====");
25             System.out.println("1. Pinjam");
26             System.out.println("2. Angsur");
27             System.out.println("3. Tampil Info");
28             System.out.println("4. Exit");
29             System.out.print("Pilih kode(1-4): ");
30             int pilih= scMila.nextInt();
```

NAMA : MILA YUNITA  
KELAS/NOMOR : TI-2C/12  
NIM : 2041720027  
MATA KULIAH : PEMROGRAMAN BERBASIS OBJEK

```
31  
32     switch(pilih){  
33         case 1:  
34             System.out.println("==PINJAM==");  
35             System.out.print("Input jumlah pinjam: ");  
36             int pinjam1 = scMila.nextInt();  
37             donny.pinjam(pinjam1);  
38             System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());  
39             break;  
40         case 2:  
41             System.out.println("==ANGSUR==");  
42             System.out.print("Input jumlah angsur: ");  
43             int bayar = scMila.nextInt();  
44             donny.angsur(bayar);  
45             System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());  
46  
47             break;  
48         case 3:  
49             System.out.println("-----");  
50             donny.tampilInfo();  
51             break;  
52         case 4:  
53             System.exit(0);  
54             break;  
55     }  
56 }  
57 }while(true);
```