



NAMA : Karmila Novi Arfiana  
NIM : 2041720073  
KELAS : TI-2C  
ABSEN : 11

### Pertanyaan Percobaan

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

**Jawab :** Karena keadaan motor masih belum on, sedangkan dalam program kita bisa menambahkan kecepatan ketika kontak on

```
public void tambahKecepatan(){  
    if(kontakOn == true){  
        kecepatan += 5;  
    }  
}
```

2. Mengapa atribut kecepatan dan kontakOn diset private?

**Jawab :** untuk menyembunyikan atribut internal dari penggunaan class lain

3. Ubah class Motor sehingga kecepatan maksimalnya adalah 100!

**Jawab :**

```
public void tambahKecepatan(){  
    if(kontakOn == true){  
        kecepatan += 5;  
  
        if(kecepatan >100){  
            kecepatan = 100;  
        }  
    }  
}
```

4. Apa yang dimaksud getter dan setter?

**Jawab :** Getter adalah public method dan memiliki tipe data return, yang berfungsi untuk mendapatkan nilai dari atribut private. Sedangkan setter adalah public method yang tidak memiliki tipe data return, yang berfungsi untuk memanipulasi nilai dari atribut private.

5. Apa kegunaan dari method getSimpanan()?

**Jawab :** untuk menampilkan semua jumlah yang disimpan.

6. Method apa yang digunakan untuk menambah saldo?

**Jawab :** method setor.



NAMA : Karmila Novi Arfiana  
NIM : 2041720073  
KELAS : TI-2C  
ABSEN : 11

7. Apa yang dimaksud konstruktor?

**Jawab** : konstruktor adalah method yang memiliki nama seperti nama classnya dan tidak memiliki return value. Konstruktor dieksekusi otomatis saat instansiasi.

8. Sebutkan aturan dalam membuat konstruktor?

**Jawab** : nama konstruktor harus sama dengan nama class dan tidak memiliki return.

9. Apakah boleh konstruktor bertipe private?

**Jawab** : ya boleh.

10. Kapan menggunakan parameter dengan passing parameter?

**Jawab** : 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.

11. Apa perbedaan atribut class dan instansiasi atribut?

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

12. 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.



NAMA : Karmila Novi Arfiana  
NIM : 2041720073  
KELAS : TI-2C  
ABSEN : 11

## 1. Tugas 1

ini adalah hasil outputnya

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

## 2. 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 class EncapDemo method set age menyatakan jika newAge lebih dari 30 maka age adalah 30.

```
public void setAge(int newAge) {
    if(newAge > 30){
        age = 30;
    }
}
```

## 3. Tugas 3

```
public void setAge(int newAge) {
    if(newAge >= 18 && newAge <= 30 ){
        age = newAge;
    }
    else{
        age = 0;
        System.out.println("Umur Anda Tidak Memenuhi Syarat");
    }
}
```

```
encap.setName("James");
encap.setAge(35);
```

```
run:
Umur Anda Tidak Memenuhi Syarat
Name : James
Age : 0
BUILD SUCCESSFUL (total time: 1 second)
```

```
encap.setName("James");
encap.setAge(17);
```



NAMA : Karmila Novi Arfiana  
NIM : 2041720073  
KELAS : TI-2C  
ABSEN : 11

```
run:
Umur Anda Tidak Memenuhi Syarat
Name : James
Age : 0
BUILD SUCCESSFUL (total time: 1 second)
```

```
encap.setName("James");
encap.setAge(18);
```

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

#### 4. Tugas 4

```
public class Anggota {
    private int noKTP;
    private String nama;
    private int limitPjm;
    private int jmlPjm;

    public Anggota(int noKTP, String nama, int limitPjm) {
        this.noKTP = noKTP;
        this.nama = nama;
        this.limitPjm = limitPjm;
        this.jmlPjm = 0;
    }

    public int getNoKTP() {
        return noKTP;
    }

    public void setNoKTP(int noKTP) {
        this.noKTP = noKTP;
    }

    public String getNama() {
        return nama;
    }

    public void setNama(String nama) {
        this.nama = nama;
    }

    public int getLimitPjm() {
        return limitPjm;
    }
}
```



NAMA : Karmila Novi Arfiana  
NIM : 2041720073  
KELAS : TI-2C  
ABSEN : 11

```
public int getJmlPjm() {  
    return jmlPjm;  
}  
  
public void pinjam(int pjm){  
    if(pjm > limitPjm){  
        System.out.println("Maaf Jumlah Pinjaman Melebihi Limit!");  
    }  
    else{  
        jmlPjm += pjm;  
    }  
}  
  
public void angsur(int bayar){  
    jmlPjm -= bayar;  
}
```

```
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)
```

## 5. Tugas 5

```
public void angsur(int bayar){  
    if(bayar < (0.1 * jmlPjm)){  
        System.out.println("Maaf, angsuran harus 10% dari jumlah pinjaman");  
    }  
    else{  
        jmlPjm -= bayar;  
    }  
}  
  
System.out.println("Jumlah pinjaman saat ini: " + donny.getJmlPjm());  
System.out.println("\nMembayar angsuran 300.000");  
donny.angsur(300000);
```



NAMA : Karmila Novi Arfiana  
NIM : 2041720073  
KELAS : TI-2C  
ABSEN : 11

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

## 6. Tugas 6

```
run:
Nama Anggota: Donny
Limit Pinjaman: 5000000
pinjam : 10000000
Maaf Jumlah Pinjaman Melebihi Limit!
Jumlah pinjaman saat ini: 0
pinjam : 4000000
Jumlah pinjaman saat ini: 4000000
Angsur : 300000
Maaf, angsuran harus 10% dari jumlah pinjaman
Jumlah pinjaman saat ini: 4000000
Angsur : 3000000
Jumlah pinjaman saat ini: 1000000
BUILD SUCCESSFUL (total time: 40 seconds)
```

```
import java.util.Scanner;
public class TestKoperasi {
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        int pinjam,bayar;

        Anggota donny = new Anggota(111333444, "Donny", 5000000);

        System.out.println("Nama Anggota: " + donny.getNama());
        System.out.println("Limit Pinjaman: " + donny.getLimitPjm());

        //System.out.println("\nMeminjam uang 10.000.000...");
        System.out.print("pinjam : ");
        pinjam = input.nextInt();
        donny.pinjam(pinjam);
        System.out.println("Jumlah pinjaman saat ini: " + donny.getJmlPjm());

        //System.out.println("\nMeminjam uang 4.000.000...");
        System.out.print("pinjam : ");
        pinjam = input.nextInt();
        donny.pinjam(pinjam);
        System.out.println("Jumlah pinjaman saat ini: " + donny.getJmlPjm());

        //System.out.println("\nMembayar angsuran 300.000");
        System.out.print("Angsur : ");
        bayar = input.nextInt();
        donny.angsur(bayar);
        System.out.println("Jumlah pinjaman saat ini: " + donny.getJmlPjm());

        //System.out.println("\nMembayar angsuran 3.000.000");
        System.out.print("Angsur : ");
        bayar = input.nextInt();
        donny.angsur(bayar);
        System.out.println("Jumlah pinjaman saat ini: " + donny.getJmlPjm());
    }
}
```