Tugas Praktikum 3

Enkapsulasi



Rafi Ody Prasetyo  
(2341720180)

D-IV Teknik Informatika  
Politeknik Negeri Malang  
Semester 3  
2024

**Percobaan 1 & 2**

Code:

package Jobsheet3.MotorEncapsulation;

public class Motor {

// Percobaan 1:

// public int kecepatan = 0;

// public boolean kontakOn = false;

private int kecepatan = 0;

private boolean kontakOn = false;

public void nyalakanMesin() {

kontakOn = true;

}

public void matikanMesin() {

kontakOn = false;

kecepatan = 0;

}

public void tambahKecepatan() {

if (kontakOn == true) {

kecepatan += 5;

} else {

System.out.println("Kecepatan tidak bisa bertambah karenam Mesin off! \n");

}

}

public void kurangiKecepatan() {

if (kontakOn == true) {

kecepatan -= 5;

} else {

System.out.println("Kecepatan tidak bisa berkurang karena mesin off! \n");

}

}

public void printStatus() {

if (kontakOn == true) {

System.out.println("Kontak On");

} else {

System.out.println("Kontak Off");

}

System.out.println("Kecepatan: "+ kecepatan +"\n");

}

}

Output:

package Jobsheet3.MotorEncapsulation;

public class MotorDemo {

public static void main(String[] args) {

Motor motor = new Motor();

motor.printStatus();

motor.tambahKecepatan();

motor.nyalakanMesin();

motor.printStatus();

motor.tambahKecepatan();

motor.printStatus();

motor.tambahKecepatan();

motor.printStatus();

motor.tambahKecepatan();

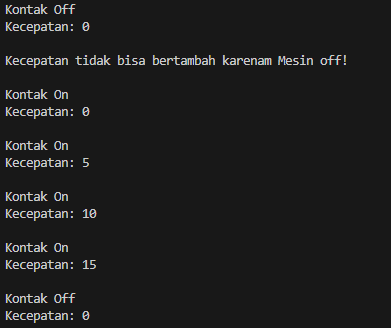
motor.printStatus();

motor.matikanMesin();

motor.printStatus();

}

}



**Pertanyaan**

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

**Jawab:**

Karena dalam method tambahKecepatan() terdapat statement if dengan syarat apabila kontakOn == True maka kecepatan dapat ditambahkan, namun apabila program tidak memenuhi syarat tersebut maka kecepatan tidak dapat bertambah karena kontakOn == False.

1. Mengapat atribut kecepatan dan kontakOn diset private?

**Jawab:**

Karena variabel tersebut hanya dibutuhkan pada class Motor.

1. Ubah class Motor sehingga kecepatan maksimalnya adalah 100!

**Jawab:**

Code:

public void tambahKecepatan() {

if (kontakOn == true) {

if (kecepatan >= 100) {

System.out.println("Kecepatan sudah maksimal! \n");

} else {

kecepatan += 50;

}

} else {

System.out.println("Kecepatan tidak bisa bertambah karenam Mesin off! \n");

}

}

Pada method tambahKecepatan() ditambahkan nested if agar ketika kecepatan mencapai 100 program akan mengeksekusi print kecepatan sudah mencapai maksimal.

**Percobaan 3 & 4**

Code:

public class Anggota {

private String nama;

private String alamat;

private float simpanan;

Anggota(String nama, String alamat) {

this.nama = nama;

this.alamat = alamat;

simpanan = 0;

}

public void setNama(String nama) {

this.nama = nama;

}

public void setAlamat(String alamat) {

this.alamat = alamat;

}

public String getNama() {

return nama;

}

public String getAlamat() {

return alamat;

}

public float getSimpanan() {

return simpanan;

}

public void setor(float uang) {

simpanan += uang;

}

public void pinjam(float uang) {

simpanan -= uang;

}

}

Output:

public class KoperasiDemo {

public static void main(String[] args) {

Anggota anggota1 = new Anggota("Iwan", "Jalan Rawar");

System.out.println("Simpanan "+ anggota1.getNama() +" : Rp. "+ anggota1.getSimpanan());

anggota1.setNama("Iwan Setiawan");

anggota1.setAlamat("Jalan Sukarno Hatta no 10");

anggota1.setor(100000);

System.out.println("Simpanan "+ anggota1.getNama() +" : Rp. "+ anggota1.getSimpanan());

anggota1.pinjam(5000);

System.out.println("Simpanan "+ anggota1.getNama() +" : Rp. "+ anggota1.getSimpanan());

}

}



**Pertanyaan 3 & 4**

1. Apa yang dimaksud getter dan setter?

**Jawab:**

Getter merupakan method yang digunakan untuk mengambil nilai dari variabel, sedangkan setter merupakan method yang digunakan untuk mengatur atau mengubah nilai dari variabel.

1. Apa kegunaan dari method getSimpanan()?

**Jawab:**

Method yang digunakan untuk mengambil nilai variabel simpanan.

1. Method apa yang digunakan untuk menambah saldo?

**Jawab:**

Method setor() dapat digunakan untuk menambah saldo.

1. Apa yand dimaksud konstruktor?

**Jawab:**

Method default yang sudah ada sejak class terbuat.

1. Sebutkan aturan dalam membuat konstruktor?

**Jawab:**

* Nama konstruktor dibuat sama dengan nama class.
* Tidak ada nilai kembalian pada konstruktor.
* Konstruktor default dibuat otomatis.

1. Apakah boleh konstruktor bertipe private?

**Jawab:**

Boleh, konstruktor privat akan ada untuk mencegah kelas tersebut dibuat.

1. Kapan menggunakan parameter dengan passsing parameter?

**Jawab:**

Menggunakan parameter dengan passing parameter dilakukan ketika sebuah metode atau konstruktor membutuhkan informasi tambahan dari luar untuk menjalankan tugasnya.

1. Apa perbedaan atribut class dan instansiasi atribut?

**Jawab:**

Instansiasi atribut dilakukan pada konstruktor, sedangkan atribut class didefinisikan diluar method apapun namun masih berada di dalam class.

1. Apa perbedaan class method dan instansiasi method?

**Jawab:**

Class method dapat mengakses dan memodifikasi atribut instance, sedangkan Tidak dapat mengakses atribut instance, tetapi bisa bekerja dengan atribut kelas (class attributes).