



NAMA : YUDAS MALABI
NIM : 2041720054
KELAS : TI 2C
MATERI : PRAKTIKUM PBO

3.3 Pertanyaan

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

Jawaban :

Karena nilai awal dari variable kontakOn, bernilai false, sehingga memunculkan peringatan "Kecepatan tidak bisa bertambah karena Mesin Off"

2. Mengapa atribut kecepatan dan kontakOn diset private?

Jawaban :

Atribut kecepatan dan kontakOn di set private karena agar atribut tersebut tidak bisa diakses dari luar class, oleh karena itu untuk mengakses dan memberinya nilai harus menggunakan method setter and getter

3. Ubah class Motor sehingga kecepatan maksimalnya adalah 100!

Jawaban :

Class Motor.java

```
public void tambahKecepatan() {  
    if (kecepatan > 100) {  
        System.out.println("Kecepatan Maksimal 100!");  
    } else {  
        if (kontakOn == true) {  
            kecepatan += 5;  
        } else {  
            System.out.println("Kecepatan tidak bisa bertambah karena Mesin off! \n");  
        }  
    }  
}
```

Output :



NAMA : YUDAS MALABI
NIM : 2041720054
KELAS : TI 2C
MATERI : PRAKTIKUM PBO

```
Kontak Off
Kecepatan: 0
Kecepatan tidak bisa bertambah karena Mesin off!

Kontak On
Kecepatan: 0
Kontak On
Kecepatan: 100
Kecepatan Maksimal 100!
Kontak On
Kecepatan: 100
Kecepatan Maksimal 100!
Kontak On
Kecepatan: 100
Kontak Off
Kecepatan: 0
```

3.6 Pertanyaan

1. Apa yang dimaksud getter dan setter?

Jawaban :

Setter : adalah sebuah aksi untuk memasukkan / mengeset suatu nilai ke dalam sebuah atribut melalui sebuah method.

Getter : adalah sebuah aksi untuk mengambil suatu nilai dari suatu atribut melalui sebuah method

2. Apa kegunaan dari method getSimpanan()?

Jawaban :

Kegunaan dari method getSimpanan() adalah untuk mengambil nilai dari atribut simpanan

3. Method apa yang digunakan untuk menambah saldo?

Jawaban :

Method yang digunakan untuk menambah saldo adalah method setor

4. Apa yang dimaksud konstruktor?

Jawaban :

Method konstruktor adalah method yang dijalankan pertama kali saat instansiasi objek dari class dilakukan.



NAMA : YUDAS MALABI
NIM : 2041720054
KELAS : TI 2C
MATERI : PRAKTIKUM PBO

5. Sebutkan aturan dalam membuat konstruktor ?

Jawaban :
1. Konstruktor tidak memiliki tipe data 2. Penamaan dari konstruktor harus sama dengan nama class 3. Konstruktor tidak memiliki access modifier 4. Konstruktor bisa memiliki parameter bisa tidak

6. Apakah boleh konstruktor bertipe private?

Jawaban :
Ya, konstruktor bisa bertipe private , dan dapat diterapkan pada singleton pattern / objek tunggal. Dengan membuat konstruktor menjadi private, maka class tersebut tidak bisa dibuat instansiasi nya diluar class tersebut

7. Kapan menggunakan parameter dengan passing parameter?

Jawaban :
Menggunakan parameter dengan passing parameter jika memerlukan nilai spesifik dalam konstruktor

8. Apa perbedaan atribut class dan instansiasi atribut ?

Jawaban :
Atribut class adalah karakteristik / ciri-ciri dari tiap class , sedangkan instansiasi atribut adalah suatu behavior / aksi untuk menjadikan atribut tersebut menjadi objek agar bisa dimodifikasi (bisa mengubah / mengeset nilainya , bisa mengambil nilai dari atributnya)

9. Apa perbedaan class method dan instansiasi method ?

Jawaban :
Class method adalah behavior / tingkah laku dari tiap class, sedangkan instansiasi method adalah suatu aksi untuk menjadikan method tersebut menjadi objek agar bisa di akses, maupun dipanggil dari class lainnya.



NAMA : YUDAS MALABI
NIM : 2041720054
KELAS : TI 2C
MATERI : PRAKTIKUM PBO

Tugas

Soal 1

EncapDemo.java :

```
public class EncapDemo {  
    private String name;  
    private int age;  
  
    public String getName() {  
        return name;  
    }  
  
    public void setName(String newName) {  
        name = newName;  
    }  
  
    public int getAge() {  
        return age;  
    }  
  
    public void setAge(int newAge) {  
        if (newAge > 30) {  
            age = 30;  
        } else {  
            age = newAge;  
        }  
    }  
}
```

EncapTest.java :

```
public class EncapTest {  
    Run | Debug  
    public static void main(String[] args) {  
        EncapDemo encap = new EncapDemo();  
        encap.setName("James");  
        encap.setAge(35);  
  
        System.out.println("Name : " + encap.getName());  
        System.out.println("Age : " + encap.getAge());  
    }  
}
```



NAMA : YUDAS MALABI
NIM : 2041720054
KELAS : TI 2C
MATERI : PRAKTIKUM PBO

Output :

```
PS C:\Users\STRIX\OneDrive\Documents\NetBeansProjects\praktikum_pbo_3> & 'c:\Users\STRIX\Java\jdk\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-Dfile.encoding=UTF-8' '.EncapTest'  
Name : James  
Age : 30
```

Soal 2

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

Jawaban :

Karena pada method setter setAge, terdapat kondisi dimana jika umur > 30 , maka nilai dari umur akan menjadi 30. Karena nilai yang di set adalah 35, maka $35 > 30$ adalah true , sehingga nilai umur akan menjadi 30 .

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

Soal 3

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

Jawaban :

Dengan memberikan kondisi di dalam method setAge.



NAMA : YUDAS MALABI
NIM : 2041720054
KELAS : TI 2C
MATERI : PRAKTIKUM PBO

```
public void setAge(int newAge) {  
    if (newAge > 30) {  
        System.out.println("Nilai maksimal adalah 30");  
        age = 30;  
    } else if (newAge < 18) {  
        System.out.println("Nilai minimal adalah 18");  
        age = 18;  
    } else {  
        age = newAge;  
    }  
}
```

Soal 4

Jawaban :

Anggota.java :

```
public class Anggota {  
    public int limitPinjam, jumlahPinjam;  
    public String nama, noKtp;  
  
    public Anggota(String noKtp, String nama, int limitPinjam) {  
        this.noKtp = noKtp;  
        this.nama = nama;  
        this.limitPinjam = limitPinjam;  
    }  
  
    public String getNama() {  
        return this.nama;  
    }  
  
    public int getLimitPinjaman() {  
        return this.limitPinjam;  
    }  
}
```



NAMA : YUDAS MALABI
NIM : 2041720054
KELAS : TI 2C
MATERI : PRAKTIKUM PBO

```
public int pinjam(int jumlahPinjam) {  
    if (jumlahPinjam > this.limitPinjam) {  
        System.out.println("Maaf, jumlah pinjaman melebihi limit");  
        return this.jumlahPinjam;  
    } else {  
        return this.jumlahPinjam += jumlahPinjam;  
    }  
}  
  
public int getJumlahPinjaman() {  
    return this.jumlahPinjam;  
}  
  
public int angsur(int angsuran) {  
    return this.jumlahPinjam -= angsuran;  
}  
}
```

Output :

```
ktikum_pbo_3\bin' 'Tugas.Soa4.TestKoperasi'  
Nama Anggota: Dony  
Limit Pinjaman: 5000000  
  
Meminjam uang 10.000.000...  
Maaf, jumlah pinjaman melebihi limit  
  
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
```

Soal 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".



NAMA : YUDAS MALABI
NIM : 2041720054
KELAS : TI 2C
MATERI : PRAKTIKUM PBO

Jawaban :

Modifikasi pada method angsur dengan memberikan kondisi .

```
public int angsur(int angsuran) {  
    if (angsuran < (this.jumlahPinjam * 0.1)) {  
        System.out.println("Maaf,angsuran harus 10% dari jumlah pinjaman");  
        return this.jumlahPinjam;  
    } else {  
        return this.jumlahPinjam -= angsuran;  
    }  
}
```

Output :

```
Nama Anggota: Dony  
Limit Pinjaman: 5000000  
  
Meminjam uang 10.000.000...  
Maaf, jumlah pinjaman melebihi limit  
  
Meminjam uang 4.000.000...  
Jumlah pinjaman saat ini: 4000000  
  
Membayar angsuran 100.000  
Maaf,angsuran harus 10% dari jumlah pinjaman  
Jumlah pinjaman saat ini: 4000000  
  
Membayar angsuran 3.000.000  
Jumlah pinjaman saat ini: 1000000
```

Soal 6

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

Jawaban :

Class TestKoperasi.java :



NAMA : YUDAS MALABI

NIM : 2041720054

KELAS : TI 2C

MATERI : PRAKTIKUM PBO

Run | Debug

```
public static void main(String[] args) {  
    int jumlahPinjaman, angsuran;  
    Scanner sc = new Scanner(System.in);  
    Anggota donny = new Anggota("111333444", "Dony", 5000000);  
    System.out.println("Nama Anggota: " + donny.getNama());  
    System.out.println("Limit Pinjaman: " + donny.getLimitPinjaman());  
    System.out.print("Masukkan jumlah pinjaman: ");  
    jumlahPinjaman = sc.nextInt();  
    System.out.println("\nMeminjam uang " + jumlahPinjaman);  
    donny.pinjam(jumlahPinjaman);  
    System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());  
    System.out.print("Masukan angsuran: ");  
    angsuran = sc.nextInt();  
    System.out.println("\nMembayar angsuran " + angsuran);  
    donny.angsur(angsuran);  
    System.out.println("Jumlah pinjaman saat ini: " + donny.getJumlahPinjaman());  
  
    sc.close();  
}
```

```
Nama Anggota: Dony  
Limit Pinjaman: 5000000  
Masukkan jumlah pinjaman: 1000000  
  
Meminjam uang 1000000  
Jumlah pinjaman saat ini: 1000000  
Masukan angsuran: 1000000  
  
Membayar angsuran 1000000  
Jumlah pinjaman saat ini: 0
```