Tugas Praktikum 4

Relasi Kelas



Rafi Ody Prasetyo  
(2341720180)

D-IV Teknik Informatika  
Politeknik Negeri Malang  
Semester 3  
2024

**Percobaan 1**

Code:

Processor.java

package ac.id.polinema;

public class Processor {

private String merk;

private double cache;

public Processor() {}

public Processor(String merk, double cache) {

this.merk = merk;

this.cache = cache;

}

public void setMerk(String merk) {

this.merk = merk;

}

public void setCache(double cache) {

this.cache = cache;

}

public String getMerk() {

return merk;

}

public double getCache() {

return cache;

}

public void info() {

System.out.printf("Merk Processor = %s\n", merk);

System.out.printf("Cache Memory = %.2f\n", cache);

}

}

Laptop.java

package ac.id.polinema;

public class Laptop {

private String merk;

private Processor proc;

public Laptop() {}

public Laptop(String merk, Processor proc) {

this.merk = merk;

this.proc = proc;

}

public void setMerk(String merk) {

this.merk = merk;

}

public void setProc(Processor proc) {

this.proc = proc;

}

public String getMerk() {

return merk;

}

public Processor getProc() {

return proc;

}

public void info() {

System.out.println("Merk Laptop = " + merk);

proc.info();

}

}

MainPercobaan1.java

package ac.id.polinema;

public class MainPercobaan1 {

public static void main(String[] args) {

Processor p = new Processor("Intel i5", 3);

Laptop l = new Laptop("Thinkpad", p);

l.info();

Processor p1 = new Processor();

p1.setMerk("Intel i5");

p1.setCache(4);

Laptop L1 = new Laptop();

L1.setMerk("Thinkpad");

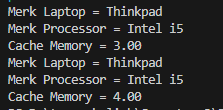
L1.setProc(p1);

L1.info();

}

}

Output:



**Pertanyaan Percobaan 1**

1. Di dalam class Processor dan class Laptop , terdapat method setter dan getter untuk masing‐masing atributnya. Apakah gunanya *method setter* dan *getter* tersebut ?

**Jawab:**

Setter digunakan untuk memasukkan nilai ke dalam variabel, sedangkan getter digunakan untuk mengambil nilai dari variabel.

1. Di dalam class Processor dan class Laptop, masing‐masing terdapat konstruktor default dan konstruktor berparameter. Bagaimanakah beda penggunaan dari kedua jenis konstruktor tersebut ?

**Jawab:**

Konstruktor default berfungsi untuk memberikan nilai awal kepada properti objek secara default ketika objek tersebut dibuat. Sedangkan konsturktor berparameter digunakan untuk memberikan nilai yang spesifik untuk properti saat objek dibuat.

1. Perhatikan class Laptop, di antara 2 atribut yang dimiliki (merk dan proc), atribut manakah yang bertipe object ?

**Jawab:**

Proc merupakan atribut yang bertipe object. Atribut ini digunakan untuk mengambil nilai dari class Processor.

1. Perhatikan class Laptop, pada baris manakah yang menunjukan bahwa class Laptop memiliki relasi dengan class Processor ?

**Jawab:**

Pada method info(), karena pada method tersebut data dari class processor ditampilkan pada method info() yang terdapat pada class laptop.

1. Perhatikan pada class Laptop , Apakah guna dari sintaks proc.info() ?

**Jawab:**

Untuk menampilkan merek laptop dan juga informasi dari processor.

1. Pada class MainPercobaan1, terdapat baris kode:

Laptop l = new Laptop("Thinkpad", p);.

Apakah p tersebut ? Dan apakah yang terjadi jika baris kode tersebut diubah menjadi:

Laptop l = new Laptop("Thinkpad", new Processor("Intel i5",3));

Bagaimanakah hasil program saat dijalankan, apakah ada perubahan ?

**Jawab:**

P merupakan variabel yang digunakan untuk memanggil konstruktor processor, dan variabel tersebut digunakan untuk menampilkan informasi processor pada konstruktor laptop. Jika kode diubah seperti diatas, hasilnya akan sama seperti menggunakan variabel p.

**Percobaan 2**

Code:

Sopir.java

public class Sopir {

private String nama;

private int biaya;

public Sopir() {}

public Sopir(String nama, int biaya) {

this.nama = nama;

this.biaya = biaya;

}

public String getNama() {

return nama;

}

public int getBiaya() {

return biaya;

}

public void setNama(String nama) {

this.nama = nama;

}

public void setBiaya(int biaya) {

this.biaya = biaya;

}

Public int hitungBiayaSopir(int hari) {

return biaya \* hari;

}

}

Mobil.java

public class Mobil {

private String merk;

private int biaya;

public Mobil() {}

public Mobil(String merk, int biaya) {

this.merk = merk;

this.biaya = biaya;

}

public void setMerk(String merk) {

this.merk = merk;

}

public String getMerk() {

return merk;

}

public void setBiaya(int biaya) {

this.biaya = biaya;

}

public int getBiaya() {

return biaya;

}

public int hitungBiayaMobil(int hari) {

return biaya \* hari;

}

}

Pelanggan.java

public class Pelanggan {

private String nama;

private Mobil mobil;

private Sopir sopir;

private int hari;

public Pelanggan() {}

public Pelanggan(String nama, Mobil mobil, Sopir sopir, int hari) {

this.nama = nama;

this.mobil = mobil;

this.sopir = sopir;

this.hari = hari;

}

public String getNama() {

return nama;

}

public void setNama(String nama) {

this.nama = nama;

}

public int getHari() {

return hari;

}

public void setHari(int hari) {

this.hari = hari;

}

public Sopir getSopir() {

return sopir;

}

public void setSopir(Sopir sopir) {

this.sopir = sopir;

}

public Mobil getMobil() {

return mobil;

}

public void setMobil(Mobil mobil) {

this.mobil = mobil;

}

public int hitungBiayaTotal() {

return mobil.hitungBiayaMobil(hari) +

sopir.hitungBiayaSopir(hari);

}

}

MainPercobaan2.java

public class MainPercobaan2 {

public static void main(String[] args) {

Mobil m = new Mobil();

m.setMerk("Avanza");

m.setBiaya(350000);

Sopir s = new Sopir();

s.setNama("John Doe");

s.setBiaya(200000);

Pelanggan p = new Pelanggan();

p.setNama("Jane Doe");

p.setMobil(m);

p.setSopir(s);

p.setHari(2);

System.out.println("Biaya Total = " +

p.hitungBiayaTotal());

}

}

Output:

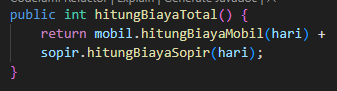


**Pertanyaan Percobaan 2**

1. Perhatikan class Pelanggan. Pada baris program manakah yang menunjukan bahwa class Pelanggan memiliki relasi dengan class Mobil dan class Sopir ?

**Jawab:**

Pada method hitungBiayaTotal(), karena disana menjumlahkan nilai hari pada class mobil dan sopir.



1. Perhatikan method hitungBiayaSopir pada class Sopir, serta method hitungBiayaMobil pada class Mobil. Mengapa menurut Anda method tersebut harus memiliki argument hari ?

**Jawab:**

Karena biaya sopir dan mobil dihitung per hari.

1. Perhatikan kode dari class Pelanggan. Untuk apakah perintah mobil.hitungBiayaMobil(hari) dan sopir.hitungBiayaSopir(hari) ?

**Jawab:**

Untuk menghitung biaya total.

1. Perhatikan class MainPercobaan2. Untuk apakah sintaks p.setMobil(m) dan p.setSopir(s) ?

**Jawab:**

Digunakan untuk memanggil data yang ada di class mobil dan sopir.

1. Perhatikan class MainPercobaan2. Untuk apakah proses p.hitungBiayaTotal() tersebut ?

**Jawab:**

Untuk menampilkan hasil dari method hitungBiayaTotal().

1. Perhatikan class MainPercobaan2, coba tambahkan pada baris terakhir dari method main dan amati perubahan saat di‐run!

System.out.println(p.getMobil().getMerk());

Jadi untuk apakah sintaks p.getMobil().getMerk() yang ada di dalam method main tersebut?

**Jawab:**

****

Digunakan untuk menampilkan nilai merek pada class Mobil.

**Percobaan 3**

Code:

Pegawai.java

public class Pegawai {

private String nip;

private String nama;

public Pegawai() {}

public Pegawai(String nip, String nama) {

this.nip = nip;

this.nama = nama;

}

public String getNip() {

return nip;

}

public void setNip(String nip) {

this.nip = nip;

}

public String getNama() {

return nama;

}

public void setNama(String nama) {

this.nama = nama;

}

public String info() {

String info = "";

info += "Nip: " + this.nip + "\n";

info += "Nama: " + this.nama + "\n";

return info;

}

}

KeretaApi.java

public class KeretaApi {

private String nama;

private String kelas;

private Pegawai masinis;

private Pegawai asisten;

public KeretaApi(String nama, String kelas, Pegawai masinis) {

this.nama = nama;

this.kelas = kelas;

this.masinis = masinis;

}

public KeretaApi(String nama, String kelas, Pegawai masinis, Pegawai asisten) {

this.nama = nama;

this.kelas = kelas;

this.masinis = masinis;

this.asisten = asisten;

}

public String getNama() {

return this.nama;

}

public String getMasinis() {

return this.masinis.getNama();

}

public String getKelas() {

return this.kelas;

}

public String getAsisten() {

return this.asisten;

}

public void setAsisten(Pegawai asisten) {

this.asisten = asisten;

}

public void setMasinis(Pegawai masinis) {

this.masinis = masinis;

}

public void setKelas(String kelas) {

this.kelas = kelas;

}

public void setNama(String nama) {

this.nama = nama;

}

public String info() {

String info = "";

info += "Nama: " + this.nama + "\n";

info += "Kelas: " + this.kelas + "\n";

info += "Masinis: " + this.masinis.info() + "\n";

info += "Asisten: " + this.asisten.info() + "\n";

return info;

}

}

MainPercobaan3.java

public class MainPercobaan3 {

public static void main(String[] args) {

Pegawai masinis = new Pegawai("1234", "Spongebob Squarepants");

Pegawai asisten = new Pegawai("4567", "Patrick Star");

KeretaApi keretaApi = new KeretaApi("Gaya Baru", "Bisnis",

masinis, asisten);

System.out.println(keretaApi.info());

}

}

**Pertanyaan Percobaan 3**

1. Di dalam method info() pada class KeretaApi, baris this.masinis.info() dan this.asisten.info() digunakan untuk apa ?

**Jawab:**

Untuk menampilkan data nip dan nama masinis dan asisten.

1. Buatlah main program baru dengan nama class MainPertanyaan pada package yang sama. Tambahkan kode berikut pada method main() !
2. Apa hasil output dari main program tersebut ? Mengapa hal tersebut dapat terjadi ?

**Jawab:**

Hasilnya akan error, karena pada kode diatas kurang menambahkan data dari asisten agar method info dapat dijalankan.

1. Perbaiki class KeretaApi sehingga program dapat berjalan !

**Jawab:**

Menambahkan data asisten.