

LAPORAN RESMI

Praktikum 2 Pengenalan Pemrograman Berbasis Obyek

Mata Kuliah: Praktek Pemrograman Berbasis Objek



Disusun oleh:

M. Ainur Ramadhan (3122500047)

2 D3 Teknik Informatika B

Dosen Pengampu: Yanuar Risah Prayogi S.Kom., M.Kom.

**PROGRAM STUDI D3 TEKNIK INFORMATIKA
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA**

2023/2024

A. TUGAS PENDAHULUAN

1. Apakah yang dimaksud dengan kelas, method, atribut dan obyek?

Jawab :

- **Kelas (Class):** cetak biru atau cetak biru dari objek. Kelas digunakan untuk membuat kerangka dasar atau resolusi dari objek yang akan digunakan. Kelas berisi definisi variabel (atribut) dan fungsi (metode) yang menggambarkan sebuah objek.
- **Metode (Method):** fungsi atau prosedur yang didefinisikan dalam sebuah kelas. Metode merupakan serangkaian perintah atau baris program yang menangani tugas tertentu. Metode dapat mengubah nilai atribut dari suatu objek dan melakukan tindakan tertentu. Metode didefinisikan pada kelas tetapi dipanggil melalui objek.
- **Atribut (Attribute):** variabel yang dideklarasikan di dalam sebuah kelas. Atribut menyimpan data atau informasi tentang objek yang dibuat berdasarkan kelas tersebut. Atribut juga dapat disebut sebagai properti.
- **Objek (Object):** hasil cetakan atau instansi dari sebuah kelas. Objek merupakan representasi nyata dari kelas yang memiliki atribut dan dapat melakukan tindakan melalui metode yang didefinisikan dalam kelas tersebut. Objek dibuat berdasarkan definisi kelas dengan menggunakan kata kunci .

2. Buatlah contoh suatu kelas dan definisikan atribut dan methodnya!

Jawab :

Class : Dosen

Atribut : nama dan nip

Method : info ()

3. Buatlah kode program soal no. 2 diatas!

Jawab :

```
public class dosen {
    public String nama;
    public int nip;
    public void info() {
        System.out.println("ini dosen PENS");
    }
}
```

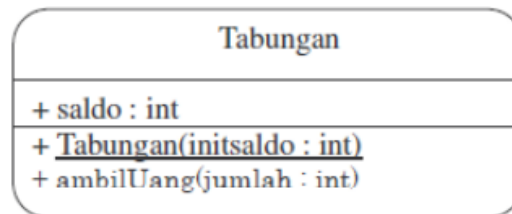
4. Buatlah kelas yang berisi main method yang membuat obyek dari kelas yang telah dibuat di soal no. 3. Selanjutnya obyek tersebut mengakses atribut dan methodnya.

Jawab :

```
public class IsiData {
    public static void main(String args[]) {
        Dosen it = new Dosen();
        it.nip = 211;
        it.nama = "Rama";
        it.info();
    }
}
```

B. LATIHAN

Latihan 1 : Mengimplementasikan UML class diagram dalam program untuk class Tabungan.



Source Code :

Tabungan.java

```
Praktikum > Praktikum 2 > Latihan 1 > J Tabungan.java > ...
1  public class Tabungan {
2      public int saldo;
3      public Tabungan (int initsaldo){
4          saldo = initsaldo;
5      }
6      public void ambilUang (int jumlah){
7          saldo = saldo - jumlah;
8      }
9  }
10
```

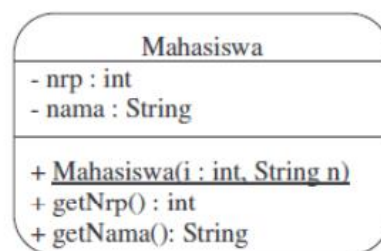
TesLatihan1.java

```
Praktikum > Praktikum 2 > Latihan 1 > J TesLatihan1.java > ...
1  public class TesLatihan1 {
2      Run | Debug
3      public static void main(String args[]) {
4          Tabungan tabungan = new Tabungan(initsaldo:5000);
5          System.out.println("Saldo awal : " + tabungan.saldo);
6          tabungan.ambilUang(jumlah:2300);
7          System.out.println(x:"Jumlah uang yang diambil : 2300");
8          System.out.println("Saldo sekarang : " + tabungan.saldo);
9      }
10 }
```

Output

```
C:\Users\USER\OOP\Praktikum\Praktikum 2\Latihan 1>javac *.java
C:\Users\USER\OOP\Praktikum\Praktikum 2\Latihan 1>java TesLatihan1
Saldo awal : 5000
Jumlah uang yang diambil : 2300
Saldo sekarang : 2700
C:\Users\USER\OOP\Praktikum\Praktikum 2\Latihan 1>
```

Latihan 2 : Mengimplementasikan UML class diagram dalam program untuk class Mahasiswa



Source Code :

Mahasiswa.java

```
Praktikum > Praktikum 2 > Latihan 2 > J Mahasiswa.java > Mahasiswa > nama
1 public class Mahasiswa {
2     public int nrp;
3     public String nama;
4     public Mahasiswa(int i, String n){
5         nrp = i;
6         nama = n;
7     }
8     public int getNrp(){
9         return nrp;
10    }
11    public String getNama(){
12        return nama;
13    }
14 }
```

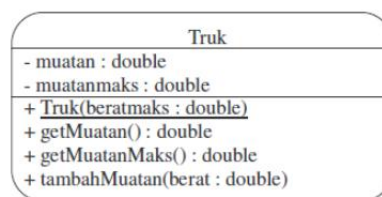
TesLatihan2.java

```
Praktikum > Praktikum 2 > Latihan 2 > J TesLatihan2.java > ...
1 public class TesLatihan2 {
2     Run | Debug
3     public static void main(String args[]){
4         Mahasiswa mhs = new Mahasiswa(i:12345, n:"Jono");
5         System.out.println("NRP : " + mhs.getNrp());
6         System.out.println("Nama : " + mhs.getNama());
7     }
8 }
```

Output

```
C:\Users\USER\OOP\Praktikum\Praktikum 2\Latihan 2> javac *.java
C:\Users\USER\OOP\Praktikum\Praktikum 2\Latihan 2> java TesLatihan2
NRP : 12345
Nama : Jono
C:\Users\USER\OOP\Praktikum\Praktikum 2\Latihan 2>_
```

Latihan 3 : Mengimplementasikan UML class diagram dalam program untuk class Truk



Source Code : Truk.java

```
Praktikum > Praktikum 2 > Latihan 3 > Truk.java > Truk > tambahMuatan(double)
1 public class Truk {
2     double muatan;
3     double muatanmaks;
4     public Truk(double beratmaks){
5         muatanmaks += beratmaks;
6     }
7     public double getMuatan(){
8         return muatan;
9     }
10    public double getMuatanMaks(){
11        return muatanmaks;
12    }
13    public void tambahMuatan( double berat){
14        if(muatan + berat <= muatanmaks)
15            muatan += berat;
16    }
17 }
```

TesLatihan3.java

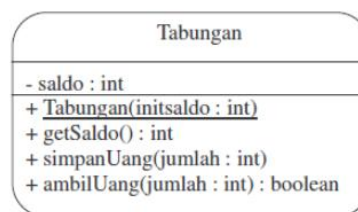
```
Praktikum > Praktikum 2 > Latihan 3 > TesLatihan3.java > TesLatihan3
1 public class TesLatihan3 {
2     Run | Debug
3     public static void main(String args[] ) {
4         Truk truk = new Truk(beratmaks:1000);
5         System.out.println("Muatan maksimal = " + truk.getMuatanMaks());
6         truk.tambahMuatan(berat:500.0);
7         System.out.println(x:"Tambah muatan : 500 ");
8         truk.tambahMuatan(berat:350.0);
9         System.out.println(x:"Tambah muatan : 350 ");
10        truk.tambahMuatan(berat:100.0);
11        System.out.println(x:"Tambah muatan : 100 ");
12        truk.tambahMuatan(berat:150.0);
13        System.out.println(x:"Tambah muatan : 150 ");
14        System.out.println("Muatan sekarang = " + truk.getMuatan());
15    }
```

Output

```
C:\Users\USER\OOP\Praktikum\Praktikum 2\Latihan 3>javac *.java
C:\Users\USER\OOP\Praktikum\Praktikum 2\Latihan 3>java TesLatihan3
Muatan maksimal = 1000.0
Tambah muatan : 500
Tambah muatan : 350
Tambah muatan : 100
Tambah muatan : 150
Muatan sekarang = 950.0
```

C. TUGAS

Tugas 1 : Mengimplementasikan UML class diagram dalam program untuk class Tabungan



Source Code :

Tabungan.java

```
Praktikum > Tugas 1 > J Tabungan.java > ...
1  public class Tabungan {
2      public int saldo;
3      public Tabungan(int initsaldo){
4          saldo = initsaldo;
5      }
6      public int getSaldo(){
7          return saldo;
8      }
9      public void simpanUang (int jumlah){
10         saldo += jumlah;
11     }
12     public boolean ambilUang (int jumlah){
13         if (saldo >= jumlah){
14             saldo -= jumlah;
15             return true;
16         } else {
17             return false;
18         }
19     }
20 }
```

TesTugas1.java

```
Praktikum > Tugas 1 > J TesTugas1.java > TesTugas1
1  public class TesTugas1 {
    Run | Debug
2  public static void main(String srt[]) {
3      boolean status;
4      Tabungan tabungan = new Tabungan(initsaldo:5000);
5      System.out.println("Saldo awal : " + tabungan.getSaldo());
6      tabungan.simpanUang(jumlah:3000);
7      System.out.println(x:"Jumlah uang yang disimpan : 3000");
8      status = tabungan.ambilUang(jumlah:6000);
9      System.out.println(x:"Jumlah uang yang diambil : 6000");
10     if (status)
11         System.out.println(x:"Ok");
12     else
13         System.out.println(x:"Gagal");
14     tabungan.simpanUang(jumlah:3500);
15     System.out.println(x:"Jumlah uang yang disimpan :3500");
16     status = tabungan.ambilUang(jumlah:4000);
17     System.out.println(x:"Jumlah uang yang diambil : 4000");
18     if (status)
19         System.out.println(x:"Ok");
20     else
21         System.out.println(x:"Gagal");
22     status = tabungan.ambilUang(jumlah:1600);
23     System.out.println(x:"Jumlah uang yang diambil : 1600");
24     if (status)
25         System.out.println(x:"Ok");
26     else
27         System.out.println(x:"Gagal");
28     tabungan.simpanUang(jumlah:2000);
29     System.out.println(x:"Jumlah uang yang disimpan :2000");
30     System.out.println("Saldo sekarang = " + tabungan.getSaldo());
31 }
32 }
```

Output

```
C:\Users\USER\OOP\Praktikum\Tugas 1>javac *.java
C:\Users\USER\OOP\Praktikum\Tugas 1>java TesTugas1
Saldo awal :5000
Jumlah uang yang disimpan : 3000
Jumlah uang yang diambil : 6000
Ok
Jumlah uang yang disimpan :3500
Jumlah uang yang diambil : 4000
Ok
Jumlah uang yang diambil : 1600
Gagal
Jumlah uang yang disimpan :2000
Saldo sekarang = 3500
C:\Users\USER\OOP\Praktikum\Tugas 1>
```

Analisa :

1. Pertama saya membuat class Tabungan pada file Tabungan.java
2. Kedua saya mendeklarasikan variable saldo dengan tipe data integer yang sesuai dengan UML diagram class Tabungan diatas

3. Pada baris kode 3 – 5 saya membuat method Tabungan yang memiliki parameter berupa initsaldo yang memiliki tipe data integer, method inii untuk menginisialisasi atau untuk mengubah nilai dari variable saldo.
4. Pada baris kode 6 – 8 saya membuat method getSaldo untuk mendapatkan nilai saldo yang telah diinisialisasi.
5. Pada baris kode 9 – 11 saya membuat method simpanUang dengan parameter jumlah yang bertipe data integer, method ini digunakan untuk menyimpan uang serta mengubah nilai saldo sebelumnya dengan menambahkan uang yang baru disimpan.
6. Pada baris kode 12 – 19 saya membuat method ambilUang dengan tipe data Boolean yang memiliki parameter jumlah dengan tipe data integer, setelah itu terdapat statement if else untuk mengimplementasikan tipe data Boolean yang akan digunakan. Statetment tersebut memiliki arti apabila saldo >= jumlah maka saldo akan dikurangi yaitu dengan saldo -= jumlah, setelah itu aka nada return untuk nilai dari saldo yang sekarang dan jika tidak memenuhi statement tersebut maka mengembalikan nilai false

Tugas 2 : : Menganalisa, membuat UML class diagram dan implementasi program

Tabel 7.1. Data karakteristik mobil

Obyek	A	B	C	D	E
mobil1	Toyota	Biru	minibus	2000	7
mobil2	Daihatsu	Hitam	pick up	1500	2
mobil3	Suzuki	Silver	suv	1800	5
mobil4	Honda	Merah	sedan	1300	5

- a. Penjelasan nama yang diidentifikasi (A, B, C, D, dan E)

A = Merk Mobil

B = Warna Mobil

C = Jenis Mobil

D = Harga Rental

E = Jumlah Sit / Tempat Duduk

- b. UML class diagram Mobil

Mobil
<ul style="list-style-type: none"> • Merk_Mobil (String) • Warna_Mobil (String) • Jenis_Mobil (String) • Harga_Rental (int) • Jumlah_Sit(int)

- **Mobil**(Merk_Mobil(String), Warna_Mobil(String), Jenis_Mobil(String), Harga_Rental(int), Jumlah_Sit(int))
- void **InfoMobil** ()

c. Class Mobil.java

```
Praktikum > Tugas 2 > J Mobil.java > ...
1  public class Mobil {
2      public String Merk_Mobil;
3      public String Warna_Mobil;
4      public String Jenis_Mobil;
5      public int Harga_Rental;
6      public int Jumlah_Sit;
7      public Mobil(String Merk_Mobil, String Warna_Mobil, String Jenis_Mobil, int Harga_Rental, int Jumlah_Sit){
8          this.Merk_Mobil = Merk_Mobil;
9          this.Warna_Mobil = Warna_Mobil;
10         this.Jenis_Mobil = Jenis_Mobil;
11         this.Harga_Rental = Harga_Rental;
12         this.Jumlah_Sit = Jumlah_Sit;
13     }
14     public void InfoMobil () {
15         System.out.println("Merk: " + Merk_Mobil);
16         System.out.println("Warna: " + Warna_Mobil);
17         System.out.println("Jenis: " + Jenis_Mobil);
18         System.out.println("Harga: " + Harga_Rental);
19         System.out.println("Jumlah Sit: " + Jumlah_Sit);
20     }
21 }
```

d. Class TesMobil.java

```
Praktikum > Tugas 2 > J TesTugas2.java > ...
1  public class TesTugas2 {
2      Run | Debug
3      public static void main (String argst[]){
4          Mobil mobil1 = new Mobil(A:"Toyota",B:"Biru",C:"Minibus",D:2000,E:7);
5          System.out.println(x:"mobil1");
6          mobil1.InfoMobil();
7          Mobil mobil2 = new Mobil(A:"Daihatsu",B:"Hitam",C:"Pick Up",D:1500,E:2);
8          System.out.println(x:"mobil2");
9          mobil2.InfoMobil();
10         Mobil mobil3 = new Mobil(A:"Suzuki",B:"Silver",C:"Suv",D:1800,E:5);
11         System.out.println(x:"mobil3");
12         mobil3.InfoMobil();
13         Mobil mobil4 = new Mobil(A:"Honda",B:"Merah",C:"Sedan",D:1300,E:5);
14         System.out.println(x:"mobil4");
15         mobil4.InfoMobil();
16     }
17 }
```

Output

```
C:\Users\USER\OOP\Praktikum\Tugas 2>javac *.java
C:\Users\USER\OOP\Praktikum\Tugas 2>java TesMobil
mobil1
Merk: Toyota
Warna: Biru
Jenis: Minibus
Harga: 2000
Jumlah Sit: 7
mobil2
Merk: Daihatsu
Warna: Hitam
Jenis: Pick Up
Harga: 1500
Jumlah Sit: 2
mobil3
Merk: Suzuki
Warna: Silver
Jenis: Suv
Harga: 1800
Jumlah Sit: 5
mobil4
Merk: Honda
Warna: Merah
Jenis: Sedan
Harga: 1300
Jumlah Sit: 5
```

Analisa :

Mobil.java

1. Pertama membuat class Mobil
2. Baris 2 – 6 Membuat variable / atribut yang sesuai dengan UML class diagram diatas
3. Baris 7 – 13 Membuat method Mobil yang memiliki parameter Mobil(String Merk_Mobil, String Warna_Mobil, String Jenis_Mobil, int Harga_Rental, int Jumlah_Sit) yang digunakan untuk mengambil data yang dimasukkan
4. Baris 14 – 20 Membuat method InfoMobil dengan tipe data void yang berguna untuk menampilkan semua info karakteristik mobil

TesMobil.java

1. Pertama membuat class TesMobil.java
2. Selanjutnya membuat public static void main(String args[]) untuk memasuki program yang akan dieksekusi
3. Baris 3 – 14 menambahkan object mobil1, mobil2, mobil3, dan mobil 4 sekaligus menambahkan karakteristik dari tiap object serta menampilkannya.

KESIMPULAN

Pada praktikum kali ini merupakan pengenalan pemrograman berbasis object, yang mana terdapat class, atribut, method dan object. Ke empat hal tersebut merupakan dasaran pada pemrograman berbasis object yang dapat digunakan untuk membuat sebuah program dengan lebih specific.