

# **PEMROGRAMAN BERORIENTASI OBJEK**

**NINF615**

**SEMESTER GASAL 2016/2017**

**MODUL PRAKTIKUM**

**PEMROGRAMAN**

**BERORIENTASI OBJEK**

**DISUSUN OLEH:**

**Tim Asisten Praktikum**

**Jurusan Teknik Elektro**

**UM**

**JURUSAN TEKNIK ELEKTRO**

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

	<b>PRAKTIKUM</b> <b>PEMROGRAMAN BERORIENTASI</b> <b>OBJEK</b>	<b>P-02</b>	<b>Pengenalan</b> <b>JAVA II</b>
	<b>KODE MATAKULIAH : NINF615</b>	<b>SEMESTER : GASAL 2016/2017</b>	

## A. Tujuan

Tujuan dari praktikum PBO ini adalah untuk:

1. Mahasiswa mampu membuat *class*.
2. Mahasiswa mampu mendeklarasikan atribut dan *method* pada *class*.
3. Mahasiswa mampu menerapkan *overloaded method*.

## B. Dasar Teori

### a) Class

Class merupakan suatu wadah yang didalamnya akan tersimpan objek-objek yang akan digunakan dalam pemrograman yang berisikan data dan fungsi yang dimiliki oleh objek tersebut. Class ini dibuat pertama kali untuk pembuatan pemrograman berbasis objek. Bagian dari class, diantaranya:

#### 1) Variabel/properti/atribut

Atribut merupakan karakteristik yang dimiliki oleh objek. Sehingga dapat mendeklarasikan atribut-atribut yang dibutuhkan kelas untuk membuat objek.

Class dapat berisi salah satu jenis variabel berikut:

- Variabel lokal : Variabel didefinisikan didalam method konstruktor atau blok disebut variabel lokal, variabel akan dideklarisasikan dalam method dan variabel akan dihancurkan ketika method telah selesai.
- Variabel Instance : Variabel dalam class tetapi diluar method apapun, variabel ini dipakai ketika class dimuat. Variabel instance dapat diakses dari dalam method, konstruktor atau class tertentu.
- Variabel Class : Variabel dideklarasikan dalam class, diluar method apapun, dengan kata kunci static.

Variabel dan metode dapat memiliki salah satu sifat berikut :

- Private, tidak dapat dipanggil dari luar class yang bersangkutan
- Protected, hanya dapat dipanggil oleh class yang bersangkutan dan anak-anak yang mewarisinya
- Public, dapat dipanggil oleh siapa saja.

## 2) Method

Method merupakan kumpulan program yang memiliki nama. Method ini bisa berbentuk seperti fungsi ataupun prosedur.

Dalam java terdapat dua buah metode (method) yaitu :

- Fungsi, merupakan metode yang memiliki nilai balik jika metode tersebut dipanggil, cara pembuatan sebuah fungsi adalah dengan cara menentukan nilai baliknya, lalu membuat nama metodenya.
- Prosedur, merupakan metode yang tidak memiliki nilai balik, cara pembuatan prosedur sama dengan fungsi namun bedanya, nilai baliknya menggunakan kata kunci void.

## 3) Konstruktor

Konstruktor merupakan method yang fungsinya untuk inialisasi atribut-atribut instan yang akan dimiliki objek. Nama method konstruktor ini sama dengan nama kelasnya, sehingga method konstruktor akan dipanggil terlebih dulu.

Contoh dari class adalah :

```
1  public class Mahasiswa{ // menggunakan class Mahasiswa
2      private String nim; // menggunakan atribut nim dengan mode private
3      private String nama; // menggunakan atribut nama dengan mode private
4      public String getName() { // method berupa fungsi
5          return nama;
6      }
7      public void setName(String nama){ //method berupa prosedur
8          this.nama=nama;
9      }
10     public String getNim(){
11         return nim;
12     }
13     public void setNim(String nim){
14         this.nim=nim;
15     }
16     public Mahasiswa(){} // default konstruktor
17     public Mahasiswa(String nim, String nama){ // konstruktor perparameter
18         this.nim=nim;
19         this.nama=nama;
20     }
21 }
```

## b) Object

Objek adalah sebuah perangkat lunak yang berisi sekumpulan atribut dan method yang berhubungan. Objek merupakan *instance* (keturunan) dari kelas. Setiap objek dibangun dari sekumpulan data atribut untuk menjabarkan karakteristik khusus dari objek dan sekumpulan method untuk menjabarkan tingkah laku dari objek.

Satu kelas bisa memiliki beberapa objek, setiap objek memiliki sifat yang sama persis seperti yang didefinisikan dalam class tersebut.

Contoh Objek adalah:

```
1 public class MahasiswaPTI {
2     public static void main(String[] args){
3         Mahasiswa MHS = new Mahasiswa(); // merupakan objek
```

### c) Scope Variable

Variabel Scope (cakupan variabel) menentukan seberapa luas variabel dapat diakses. Ruang lingkup atau scope suatu variabel merupakan penempatan lokasi dimana variabel tersebut dideklarasikan dan dapat berlaku. Lingkup variabel terbagi dua, yaitu lingkup **Global** dan **Local**.

### d) Overloading Methods

Overloading adalah method dengan nama yang sama dengan method lain pada suatu class tetapi dengan parameter yang berbeda. Tujuan dibuatnya overloading yaitu memudahkan penggunaan method dengan fungsi yang hampir sama.

Contoh overloading method adalah:

```
1 public void tambah1(){
2     int a=5, b=10;
3     System.out.println("Hasil Pertambahan dari method tambah1 ke-1 = +(a+b));
4 }
5
6 // method tambah1 di overloading dengan 2 parameter (int x, int y)
7 public void tambah1(int x, int y){
8     System.out.println("Hasil Pertambahan dari method tambah1 ke-2 = +(x+y));
9 }
```

### e) Input/Output

Standar input output yang dapat ditangani pemrograman Java:

- System.in, berguna untuk menangani pembacaan input user dari keyboard (standar input).
- System.out, berguna untuk mengirimkan keluaran (output) ke layar (standar output), biasanya dikombinasikan dengan metode println().
- System.err, digunakan untuk mengirimkan pesan kesalahan. Biasanya diletakkan pada Exception Handling.

Berikut ini adalah langkah-langkah yang perlu dilakukan untuk mendapatkan input :

1. Tambahkan di bagian paling atas code anda:

```
import java.io.*;
```

2. Tambahkan statemen berikut pada method main:

```
BufferedReader dataIn = new BufferedReader (new  
InputStreamReader(System.in));
```

3. Deklarasikan variabel String temporary dan gunakan fungsi readLine() untuk mendapatkan input serta ketikkan pada blog *try-catch*.

```
try {  
    String temp = dataIn.readLine();  
}  
catch ( IOException e ){  
    System.out.println("Terjadi kesalahan saat anda melakukan input ")  
}
```

Berikut ini adalah langkah-langkah yang perlu dilakukan untuk mendapatkan input dari console dengan Scanner:

1. Tambahkan di bagian atas code anda:

```
import java.util.Scanner;
```

2. Membuat objek dari class Scanner.

```
Scanner input = new Scanner(System.in);
```

3. Memasukkan nilai dari masing-masing variabel dengan menggunakan object dari Scanner.

```
System.out.print("Masukan Nama : ");  
String nama = input.nextLine();  
System.out.println(nama);
```

## C. Latihan

### 1. Latihan 1

- a. Buat file Mahasiswa.java seperti berikut:

```
1 public class Mahasiswa {
2     public String nama;
3     public int angkatan;
4     public String asal;
5     public Mahasiswa(){
6         nama= "Ani";
7         angkatan = 2010;
8         asal = "Magelang";
9     }
10    public void ubah(String nm, int ak, String as){
11        nama = nm;
12        angkatan = ak;
13        asal = as;
14    }
15    public void cetak(){
16        System.out.println("nama : "+nama);
17        System.out.println("angkatan : "+angkatan);
18        System.out.println("asal : "+asal);
19    }
20 }
```

- b. Buat file TestMahasiswa

```
1 public class TestMahasiswa {
2     public static void main(String[] args){
3         Mahasiswa MHS = new Mahasiswa();
4         System.out.println("IDENTITAS AWAL : ");
5         MHS.cetak();
6
7         MHS.ubah("DITA", 2010, "Madiun");
8         System.out.println("IDENTITAS AKHIR : ");
9         MHS.cetak();
10    }
11 }
```

- c. Kompilasi kedua file tersebut.
- d. Buat 2 object baru dengan nama mhsTI dan mhsTE dalam file TestMahasiswaRevisi.java lalu ubah data kedua object.
- e. Panggil fungsi cetak.

### 2. Latihan 2

- a. Buat file Lingkup.java seperti di bawah ini:

```
1 public class Lingkup {
2     //Variabel instan
3     String sifat = "Marah";
4     //Metode
5     void infoLingkup(){
6         String sifat = "Malu";
7         System.out.println("sifat pada metode: "+sifat);
8         System.out.println("sifat milik kelas: "+this.sifat);
9     }
10 }
```

b. Buat TestLingkup.java seperti di bawah ini:

```
1 public class TestLingkup {
2     public static void main(String[] args){
3         Lingkup varx = new Lingkup();
4         varx.infoLingkup();
5     }
6 }
```

### 3. Latihan 3

a. Buat file Scoope.java seperti di bawah ini:

```
1 public class Scoope {
2     //variabel instan
3     int angka = 20;
4     //Metode
5     void infoScoope(){
6         int angka1 = 12;
7         System.out.println("angka di method: "+angka1);
8         System.out.println("angka di kelas: "+this.angka);
9         int modulus = angka % angka1;
10        System.out.println("sisa hasil: "+modulus);
11        float bagi = angka / angka1;
12        System.out.println("hasil bagi: "+bagi);
13    }
14 }
```

b. Buat TestScoope.java seperti di bawah ini:

```
1 public class TestScoope {
2     public static void main(String[] args){
3         Scoope varx = new Scoope();
4         varx.infoScoope();
5     }
6 }
```

### 4. Latihan 4

Membuat file yang memerlukan inputan. Simpan file dengan nama Report.java

```
1 import java.io.*;
2
3 public class Report {
4     public static String bacaInputan(){
5         String baca="";
6         InputStreamReader input = new InputStreamReader (System.in);
7         BufferedReader buf = new BufferedReader (input);
8         try{
9             baca = buf.readLine();
10        }catch ( IOException e ){
11            System.out.println("Error!");
12        }
13        return baca;
14    }
15    public static void main(String[] args){
16        //Digunakan untuk menangkap exception.
17        String name = "";
18        int nilaiMatematika;
19        int nilaiFisika;
20        int nilaiKimia;
```



```

21     System.out.print("Masukkan nama :");
22     name=bacaInputan();
23     System.out.print("Masukkan nilai matematika : ");
24     nilaiMatematika=Integer.parseInt(bacaInputan());
25     System.out.print("Masukkan nilai Fisika : ");
26     nilaiFisika=Integer.parseInt(bacaInputan());
27     System.out.print("Masukkan nilai Kimia : ");
28     nilaiKimia=Integer.parseInt(bacaInputan());
29
30     System.out.println("\n-----");
31     System.out.println(""+name);
32     System.out.println("Nilai kamu adalah ");
33     System.out.println("Matematika : "+nilaiMatematika);
34     System.out.println("Fisika : "+nilaiFisika);
35     System.out.println("Kimia : "+nilaiKimia);
36 }
37 }

```

## 5. Latihan 5

Buat file Persegi.java seperti di bawah ini:

```

1  import java.io.*;
2
3  public class Persegi {
4      public static void main(String[] args){
5          BufferedReader data = new BufferedReader(new InputStreamReader(System.in));
6          String x = " ";
7          int hitungLuas=0;
8          System.out.println("---Luas Persegi---");
9          System.out.print("Masukkan nilai sisi : ");
10
11         try {
12             x = data.readLine();
13             int angka = Integer.parseInt(x);
14             hitungLuas = angka * angka;
15         }catch (Exception e){
16             System.out.println("Error");
17         }
18         System.out.println("Luas persegi dengan sisi "+x+" adalah "+hitungLuas);
19     }
20 }
--

```

## D. Tugas Praktikum

1. Buat program overloading perkalian dengan potongan script berikut dan hasilnya seperti di bawah ini:

```

static int kali(int a, int b){
    return a*b;
}

static double kali(double a, double b){
    return a*b;
}

static int kali(int a,int b, int c){
    return a*b*c;
}

```

```

1. 9
2. a(1) * b(9) = 9
3. 18
4. 17.5

```

2. Buat demo kelas Boolean dengan menggunakan Statemen IF sehingga keluarannya sebagai berikut:

```

-----
Nilai b = true
Statemen awal dieksekusi
-----
Nilai b = false
Statemen akhir dieksekusi
-----
4 <= 2 mengembalikan nilai false
11 > 9 mengembalikan nilai true
5 > 5 mengembalikan nilai false

```

3. Buat sebuah program yang menggunakan konsep input/output sehingga dapat menghasilkan output sebagai berikut:

```

Please Input Your Name      : Keisya
Please Input Your Character : nice
Please Input Your Hobby     : singing

Hello...My Name is Keisya.
I'm a very nice person.
My Hobby is singing.

```

## E. Tugas Rumah

1. Apa fungsi dari kata kunci **this** ?
2. Jelaskan mengapa angka yang terdapat pada *class* dan *method* dapat dihitung ?
3. Bagaimana hasil boolean dari  $9 \geq 9$  dan  $9 \leq 9$  ? jelaskan hasilnya !
4. Buatlah program yang menghasilkan output 2 buah segitiga seperti berikut.

```

C:\>
Input bilangan : 6
Segitiga sama sisi, sisi=6
* * * * *
 * * * *
  * * *
   * *
    *

Segitiga siku-siku dengan alas=6, tinggi=6
*
* *
* * *
* * * *
* * * * *

```

5. Buatlah sebuah program absensi dimana terdapat menu :
  - a. Menambah pegawai
  - b. Melihat jumlah pegawai yang masuk
  - c. Keluar (exit)