

# BAB 3

## OVERLOAD DAN OVERLOADING METHOD

### Tujuan

1. Praktikan mampu memahami konsep static method yang ada di java
2. Mampu membedakan perbedaan method yang menggunakan kata kunci static atau tidak
3. Mampu memahami dan mengimplementasikan method overloading

### Ringkasan Materi

#### A. Overloading Method

Penamaan method pada OOP (Object Oriented Programming) menjadi sangat penting terutama pada pemrograman menggunakan bahasa java. Dalam penamaan method, terkadang tanpa sadar kita memberi nama yang sama pada method yang berbeda sehingga dapat mengakibatkan kesalahan pada saat program dijalankan. Untuk mengatasi hal ini, Java memperkenalkan istilah *overloading*, Overloading adalah teknik penamaan method dengan nama yang sama namun memiliki tipe dan jumlah argumen atau parameter yang berbeda. Sebagai contoh adalah method Hitung pada class Lingkaran, dimana pada class ini terdapat method bernama Hitung dengan parameter a dengan tipe integer.

```
public class Lingkaran{  
    public static void Hitung(int a){  
        //kode program  
    }  
}
```

Kemudian pada class tersebut dibuat method baru bernama Hitung namun parameternya bertipe double dengan nama value

```
public static void Hitung(double value){  
    //kode program  
}
```

Kedua method ini disebut overloading method karena memiliki nama yang sama tetapi tipe dari argumennya berbeda.

#### B. Overloading Constructor

Sama halnya dengan Overloading Method, Overloading Constructor juga mempunyai karakteristik yang serupa, namun hanya saja peletakannya yang berbeda yaitu pada constructornya saja.

```
public Lingkaran(int alas){  
    //kode program  
}  
  
public Lingkaran(int alas, int tinggi){  
    //kode program  
}
```

### Pelaksanaan Percobaan

#### A. Overloading Method

Ketikkan program di bawah ini

1	import java.util.Scanner;
---	---------------------------

```

2 public class Overloading {
3     public static void HitungLuas(int a,int b){
4         int nilai = a*b;
5         System.out.println("maka hasil luas : "+nilai);
6     }
7     public static double HitungLuas(double value, double value2){
8         double nilai = value* value2;
9         return nilai;
10    }
11    public static void main(String[] args) {
12        Scanner in = new Scanner(System.in);
13        System.out.print("masukkan nilai integer 1 : ");
14        int integer1 = in.nextInt();
15        System.out.print("masukkan nilai integer 2 : ");
16        int integer2 = in.nextInt();
17        HitungLuas(integer1, integer2);
18        System.out.print("masukkan nilai double 1 : ");
19        double double1 = in.nextDouble();
20        System.out.print("masukkan nilai double 2 : ");
21        double double2 = in.nextDouble();
22        HitungLuas(integer1, integer2);
23        System.out.println("Maka          hasil          luas          :
24        "+HitungLuas(double1, double2));
25    }
26 }

```

## B. Overloading Constructor

Ketikkan program dibawah ini

```

1 public class lingkaran{
2     int alas, tinggi;
3     public lingkaran(int alas){
4         this.alas = alas;
5     }
6     public lingkaran(int alas, int tinggi){
7         this.alas = alas;
8         this.tinggi = tinggi;
9     }
10    public void setAlas(int alas){
11        this.alas = alas;
12    }
13    public void setTinggi(int tinggi){
14        this.tinggi = tinggi;
15    }
16    public int getAlas(){
17        return alas;
18    }
19    public int getTinggi(){
20        return tinggi;
21    }
22    public double hitungLuas(){
23        double hasil = (double) (getTinggi()*getAlas())/2;
24        return hasil;
25    }

```

26	public void displayMessage(){
27	System.out.println("Hitung Luas : "+hitungLuas());
28	}
29	}

Ketikkan main classnya

1	public class LIngkaran
2	public
3	
4	l.setTinggi(10);
5	l.displayMessage();
6	lingkaran l2 = new lingkaran(4, 10);
7	l2.displayMessage();
8	}
9	}

## Data dan Analisis hasil percobaan

### A. Overloading Method

Pertanyaan

#### 1. Lakukan percobaan diatas dan benahi jika menemukan kesalahan!

Tidak terdapat kesalahan pada kode Overloading Method diatas

#### 2. Jika pada baris 7, pada parameter double value dan double value2 di hapus dan di gantimenjadi int a dan int b apa yang terjadi? Jelaskan!

**PENJELASAN :** Jika parameter double value dan double value2 diganti dengan int a dan int b, maka akan terjadi error pada parameter tersebut. Alasannya karena method pada baris 7 bertipe double, jadi tidak dapat menerima parameter int, kemudian untuk alasan berikutnya karena dalam java, metode overloading terjadi ketika nama metode yang sama digunakan dalam sebuah kelas dengan parameter yang berbeda, dengan menghapus parameter double value dan double value2 dan menggantinya dengan int a dan int b itu sebenarnya akan membuat method HitungLuas dengan parameter int yang persis sama dengan metode HitungLuas dengan parameter int yang persis sama dengan metode HitungLuas yang lain

#### 3. Rubah method pada baris ketujuh menjadi method bertipe void, dan lakukan juga perubahan main method.

```

Tugas4 > src > J Overloading.java > ...
1  import java.util.Scanner;
2  public class Overloading {
3      public static void HitungLuas(int a, int b) {
4          int nilai = a * b;
5          System.out.println("maka hasil luas : " + nilai);
6      }
7
8      public static void HitungLuas(double value, double value2) {
9          double nilai = value * value2;
10         System.out.println("maka hasil luas : " + nilai);
11     }
12
13     Run | Debug
14     public static void main(String[] args) {
15         Scanner in = new Scanner(System.in);
16         System.out.print(s:"masukkan nilai integer 1 : ");
17         int integer1 = in.nextInt();
18         System.out.print(s:"masukkan nilai integer 2 : ");
19         int integer2 = in.nextInt();
20         HitungLuas(integer1, integer2);
21         System.out.print(s:"masukkan nilai double 1 : ");
22         double double1 = in.nextDouble();
23         System.out.print(s:"masukkan nilai double 2 : ");
24         double double2 = in.nextDouble();
25         HitungLuas(double1, double2);
26     }

```

## B. Overloading Constructor

Pertanyaan

1. Lakukan percobaan diatas dan benahi jika menemukan kesalahan!

Tidak terdapat kesalahan pada kode Overloading Constructor tersebut

2. Pada class lingkaran Tambahkan constructor dengan parameter int tinggi, apa yang terjdadan jelaskan!

```

... ← → Nisa-Aulia-Harismadani-PBO-PTI-A
J lingkaran.java 2, U x
Tugas4 > src > J lingkaran.java > lingkaran > alas
1  public class lingkaran {
2      int alas, tinggi;
3      public lingkaran(int alas) {
4          this.alas = alas;
5      }
6
7      public lingkaran(int tinggi) {
8          this.tinggi = tinggi;
9      }
10
11     public lingkaran(int alas, int tinggi) {
12         this.alas = alas;
13         this.tinggi = tinggi;
14     }

```

**PENJELASAN :** Terjadi error karena nama parameternya sama antara konstruktor ketiga dengan konstruktor yang baru saja saya tambahkan (konstruktor kedua), hal tersebut menyebabkan compiler ambigu dalam pemilihan konstruktor, untuk mengatasi error tersebut adalah dengan memberikan nama yang berbeda pada konstruktor kedua misal (newTinggi)

3. Pada class lingkaran tambahkan constrctor dengan tipe data String alas dan String tinggi,kemudian tambahkan method untuk melakukan parsing atau perubahan tipe data dari String menjadi integer. Setelah itu pada method main lakukan instansiasi objek dengan nama objek Lstring dengan memanggil konstruktor bertipe data String. Jelaskan!

```
1 public class lingkaran {
4     public lingkaran(int alas) {
7
8     public lingkaran(int alas, int tinggi) {
9         this.alas = alas;
10        this.tinggi = tinggi;
11    }
12
13    public lingkaran(String alas, String tinggi) {
14        this.alas = parseInt(alas);
15        this.tinggi = parseInt(tinggi);
16    }
17
18    private int parseInt(String value) {
19        try {
20            return Integer.parseInt(value);
21        } catch (NumberFormatException e) {
22            System.out.println(x:"Error: Tidak dapat melakukan parsing string menjadi integer.");
23            return 0;
24        }
25    }
26 }
```

```
1 public class MainLingkaran {
2     public static void main(String[] args) {
3         lingkaran l1 = new lingkaran (alas:3);
4         l1.setTinggi (tinggi:10);
5         l1.displayMessage();
6         lingkaran l2 = new lingkaran (alas:4, tinggi:10);
7         l2.displayMessage();
8         lingkaran lstring = new lingkaran(alas:"5", tinggi:"8");
9         lstring.displayMessage();
10    }
11 }
```

**PENJELASAN :** Pada kelas lingkaran terdapat penambahan constructor baru dengan parameter String alas dan String tinggi, nah didalam constructor tersebut nilai alas dan tinggi akan diubah dari tipe data String menjadi int dengan method parseInt, dalam method parseInt menggunakan Integer.parseInt(value) untuk melakukan parsing String menjadi int, jika parseInt tersebut berhasil maka method akan mengembalikan nilai int dan jika sebaliknya maka method akan mengembalikan nilai default 0.

Kermudian pada method main dibuat objek dengan nama Lstring yang akan memanggil constructor lingkaran(String alas, String tinggi)

## Tugas Praktikum

Melanjutkan dari bab 1 dan 2, buatlah method berikut :

1. Method `bacaFile("pathFile")`  
Fungsinya adalah memberi nilai pada kelas kalian dengan nilai atribut sesuai yang ada di file txt. Apabila kelas kalian pada bab 1 memiliki atribut judul dan penulis, maka pada file txt akan ada 2 nilai untuk judul dan nilai penulis (isi txt : DILAN 1990; Pidi Baiq)
2. Method `simpanFile("namaFile")`  
Fungsinya adalah menyimpan nilai atribut kelas.
3. Method `hitungRoyalti` dengan parameter harga 1 buku, yang mengembalikan nilai 10% dari penjualan bulan itu. (1 parameter)
4. Method `hitungRoyalti` dengan parameter harga 1 buku dan nilai persen yang ingin diambil dari penjualan bulan itu. (2 parameter)