

**LAPORAN PRAKTIKUM
PEMROGRAMAN BERORIENTASI OBJECT
TUGAS JOBSHEET 7**



Sahrul Ramadhani

244107020058

2H

D-IV Teknik Informatika

Teknologi Informasi

Politeknik Negeri Malang

2025

PERCOBAAN 1

Hasil RUN:

```
PS D:\Pemrograman Berbasis Object> d:; cd 'd:\Pemrograman Berbasis Object\
etailsInExceptionMessages' '-cp' 'C:\Users\syahr\AppData\Roaming\Code\Us
dt_ws\Pemrograman Berbasis Object_b3ef0170\bin' 'Utama'
Program Testing Class Manager & Staff
Manager :Administrasi
NIP :101
Nama :Tedjo
Golongan :1
Tunjangan :5000000
Gaji :10000000
Bagian :Administrasi
-----
NIP :0003
Nama :Usman
Golongan :2
Jml Lembur :10
Gaji Lembur :10000
Gaji :3100000
NIP :0005
Nama :Anugrah
Golongan :2
Jml Lembur :10
Gaji Lembur :55000
Gaji :3550000
-----
Manager :Pemasaran
NIP :102
Nama :Atika
Golongan :1
Tunjangan :2500000
Gaji :7500000
Bagian :Pemasaran
```

```

-----
NIP :0004
Nama :Hendra
Golongan :3
Jml Lembur :15
Gaji Lembur :5500
Gaji :2082500
NIP :0006
Nama :Arie
Golongan :4
Jml Lembur :5
Gaji Lembur :100000
Gaji :1500000
NIP :0007
Nama :Mentari
Golongan :3
Jml Lembur :6
Gaji Lembur :20000
Gaji :2120000
-----
PS D:\Pemrograman Berbasis Object>

```

Kode Java terlampir pada folder yang sama dengan Laporan ini

LATIHAN

Untuk Class PerkalianKu

```

public class PerkalianKu {

    void perkalian(int a, int b){
        System.out.println(a * b);
    }

    void perkalian(int a, int b, int c){
        System.out.println(a * b * c);
    }

    public static void main(String args []){
        PerkalianKu objek = new PerkalianKu();

        objek.perkalian(25, 43);
        objek.perkalian(34, 23, 56);
    }
}

```

1. Dari source coding diatas terletak dimanakah overloading?
Ada di Method perkalian
void perkalian(int a, int b)
void perkalian(int a, int b, int c)
2. Ada 1 jumlah yang berbeda?
Method pertama int a, int b
Method kedua int a, int b, int c

Versi 2

```
public class PerkalianKu2 {  
    void perkalian (int a, int b){  
        System.out.println(a*b);  
    }  
  
    void perkalian (double a, double b){  
        System.out.println(a*b);  
    }  
  
    public static void main(String[] args) {  
        PerkalianKu2 objek = new PerkalianKu2();  
  
        objek.perkalian(25, 43);  
        objek.perkalian(34.56, 23.7);  
    }  
}
```

3. Dari source coding diatas terletak dimanakah overloading?

Pada method perkalian sprti soal sebelummnya

```
void perkalian (int a, int b)  
void perkalian (double a, double b)
```

4. Jika terdapat overloading ada berapa tipe parameter yang berbeda?

Ada 2 yang berbeda, pada method pertama menggunakan parameter bertipe data INT
Dan method kedua menggunakan parameter bertipe data Double

Pada Kode ini:

```
public class Ikan {  
    public void swim () {  
        System.out.println("Ikan bisa berenang");  
    }  
}  
  
public class Piranha extends Ikan {  
    public void swim () {  
        System.out.println("PIranha bisa makan daging");  
    }  
}  
  
public class Fish {  
    public static void main(String[] args) {  
        Ikan a = new Ikan();  
        Ikan b = new Piranha();  
  
        a.swim();  
        b.swim();  
    }  
}
```

5. Dari source coding diatas terletak dimanakah overriding?

Pada method swim di class Piranha yang mengoverriding swim di class Ikan

6. Jabarkanlah apabila sourcoding diatas jika terdapat overriding?

Overriding melakukan timpa nilai methodnya, namun Nama method harus sama, Parameter yang sama, dan Tipe pengembalian (return type) yang sama dengan method di superclass (induk).

Keduanya memiliki method yang sama namun isinya berbeda.

a.swim() memanggil method swim() milik class Ikan

b.swim() memanggil method swim() milik class Piranha (hasil overriding)

TUGAS

1. Overloading

Kode java telah saya lampirkan pada folder yang sama
Dengan nama file Segitiga.java dan SegitigaMain.java
Berikut adalah hasil RUN:

```
PS D:\bin> java SegitigaMain
Total sudut 1 Nilai Segitiga adalah = 120
Total sudut 2 Nilai Segitiga adalah = 90
Total Keliling 3 Sisi Segitiga adalah = 15
Keliling Segitiga siku 2 sisi adalah = 24.0
PS D:\Pemrograman Berbasis Object>
```

2. Override

Kode Javatelah saya lampirkan pada folder yang sama, dengan nama file Manusia.java, Dosen.java, Mahasiswa.java dan MainTugas.java

Disini Sebenarnya method yang di override hanyalah method makan();

Sedangkan pada di Output Method lembur() dan tidur() memerlukan downcasting karena bukan bagian dari superclass Manusia, sedangkan bernafas() ketika di panggil tetap tdk berubah teksnya karena milik Manusia dan tidak di-override.

```
Akses class Dosen
Manusia Sedang Bernafas
Dosen Sedang Makan
Dosen Sedang Lembur
=====
Akses class Mahasiswa
Manusia Sedang Bernafas
Mahasiswa Sedang Makan
Mahasiswa Sedang tidur
=====
Akses class Parent Manusia
Manusia Sedang Bernafas
Manusia Sedang Makan
```