

# **LAPORAN PRAKTIKUM**

## **PEMROGRAMAN BERORIENTASI**

### **OBJEK**

### **PERTEMUAN 6**



2022

**Praktikan**

2141762056

Margaretha Violina Putri Purnomo

SIB 2F /11

## DAFTAR ISI

|                                           |           |
|-------------------------------------------|-----------|
| <b>DAFTAR ISI .....</b>                   | <b>2</b>  |
| <b>1. KOMPETENSI .....</b>                | <b>3</b>  |
| <b>2. LINK GITHUB.....</b>                | <b>3</b>  |
| <b>3. PRAKTIKUM.....</b>                  | <b>3</b>  |
| 2.1 PERCOBAAN 1 – EXTENDS .....           | 3         |
| 2.1.1 Langkah-Langkah Percobaan.....      | 3         |
| 2.1.2 Pertanyaan.....                     | 4         |
| 2.2 PERCOBAAN 2 – HAK AKSES .....         | 6         |
| 2.2.1 Langkah-Langkah Percobaan.....      | 6         |
| 2.2.2 Pertanyaan.....                     | 7         |
| 2.3 PERCOBAAN 3 – SUPER .....             | 8         |
| 2.3.1 Langkah-Langkah Percobaan.....      | 8         |
| 2.3.2 Pertanyaan.....                     | 9         |
| 2.4 PERCOBAAN 4 – SUPER CONSTRUCTOR ..... | 10        |
| 2.4.1 Langkah-Langkah Percobaan.....      | 10        |
| 2.4.2 Pertanyaan.....                     | 11        |
| <b>TUGAS .....</b>                        | <b>13</b> |

## 1. Kompetensi

1. Memahami konsep dasar inheritance atau pewarisan.
2. Mampu membuat suatu subclass dari suatu superclass tertentu

## 2. Link Github

<https://github.com/MargarethaViolinaPutri/PBO->

## 3. Praktikum

### 2.1 Percobaan 1 – Extends

#### 2.1.1 Langkah-Langkah Percobaan

1. Buatlah sebuah class parent/superclass dengan nama ClassA.java

```
11 public class ClassA {
12     public int x;
13     public int y;
14
15     public void getNilai() {
16         System.out.println("nilai x:" + x);
17         System.out.println("nilai y:" + y);
18     }
19 }
```

2. Buatlah sebuah class anak/subclass dengan nama ClassB.java

```
public class ClassB {
    public int z;

    public void getNilaiZ() {
        System.out.println("nilai Z:" + z);
    }

    public void getJumlah() {
        System.out.println("jumlah:" + (x+y+z));
    }
}
```

## INHERITENCE

3. Buatlah class Percobaan1.java untuk menjalankan program diatas!

```
11 public class Percobaan1 {
12
13     public static void main(String[] args) {
14         ClassB hitung = new ClassB();
15         hitung.x = 20;
16         hitung.y = 30;
17         hitung.z = 5;
18         hitung.getNilai();
19         hitung.getNilaiZ();
20         hitung.getJumlah();
21     }
22 }
```

4. Jalankan program diatas, kemudian amati apa yang terjadi!

### COMPILATION ERROR :

```
-----
Percobaan1/Percobaan1.java:[15,15] cannot find symbol
  symbol:   variable x
  location: variable hitung of type Percobaan1.ClassB
Percobaan1/Percobaan1.java:[16,15] cannot find symbol
  symbol:   variable y
  location: variable hitung of type Percobaan1.ClassB
Percobaan1/Percobaan1.java:[18,15] cannot find symbol
  symbol:   method getNilai()
  location: variable hitung of type Percobaan1.ClassB
Percobaan1/ClassB.java:[18,40] cannot find symbol
  symbol:   variable x
  location: class Percobaan1.ClassB
Percobaan1/ClassB.java:[18,42] cannot find symbol
  symbol:   variable y
  location: class Percobaan1.ClassB
5 errors
-----
```

### BUILD FAILURE

### 2.1.2 Pertanyaan

1. Pada percobaan 1 diatas program yang dijalankan terjadi error, kemudian perbaiki sehingga program tersebut bisa dijalankan dan tidak error!

#### Jawab :

Menambahkan extends pada ClassB. Sehingga ClassB merupakan keturunan dari ClassA.

## INHERITENCE

```
11 public class ClassB extends ClassA{
12     public int z;
13
14     public void getNilaiZ(){
15         System.out.println("nilai Z:" + z);
16     }
17     public void getJumlah(){
18         System.out.println("jumlah:" + (x+y+z));
19     }
20 }
21
```

Maka pada percobaan 1 tidak error

```
-----< com.mycompany:Semester3_Retha >-----
Building Semester3_Retha 1.0-SNAPSHOT
-----[ jar ]-----

--- exec-maven-plugin:3.0.0:exec (default-cli) @ Semester3_Retha ---
nilai x:20
nilai y:30
nilai Z:5
jumlah:55

BUILD SUCCESS

Total time: 3.868 s
Finished at: 2022-10-16T17:25:18+07:00
-----
```

2. Jelaskan apa penyebab program pada percobaan 1 ketika dijalankan terdapat error!

**Jawab :** Karena pada atribut yang dipanggil pada class Percobaan1 tidak ada pada ClassB, namun ada pada ClassA. Sehingga ditambahkan extends pada ClassB terhadap ClassA, agar atribut yang menjadi milik ClassA juga milik ClassB.

## INHERITENCE

### 2.2 Percobaan 2 – Hak Akses

#### 2.2.1 Langkah-Langkah Percobaan

1. Buatlah sebuah class parent/superclass dengan nama ClassA.java

```
1 public class ClassA {  
2     private int x;  
3     private int y;  
4  
5     public void setX(int x) {  
6         this.x = x;  
7     }  
8     public void setY(int y) {  
9         this.y = y;  
10    }  
11    public void getNilai() {  
12        System.out.println("nilai x: "+x);  
13        System.out.println("nilai y: "+y);  
14    }  
15 }
```

2. Buatlah sebuah class parent/superclass dengan nama ClassB.java

```
1 public class ClassB {  
2     private int z;  
3  
4     public void setZ(int z) {  
5         this.z = z;  
6     }  
7     public void getNilaiZ() {  
8         System.out.println("nilai Z: "+z);  
9     }  
10    public void getJumlah() {  
11        System.out.println("jumlah : " + (x+y+z));  
12    }  
13 }
```

3. Buatlah class Percobaan2.java untuk menjalankan program diatas!

```
11 public class Percobaan2 {  
12     public static void main(String[] args) {  
13         ClassB hitung = new ClassB();  
14         hitung.setX(20);  
15         hitung.setY(30);  
16         hitung.setZ(5);  
17         hitung.getNilai();  
18         hitung.getNilaiZ();  
19         hitung.getJumlah();  
20     }  
21 }
```

4. Jalankan program diatas, kemudian amati apa yang terjadi!

## INHERITENCE

```

-----
BUILD FAILURE
-----
Total time: 3.452 s
Finished at: 2022-10-16T17:47:42+07:00
-----

Failed to execute goal org.codehaus.mojo:exec-maven-plugin:3.0.0:exec (default-cli) on project Semester3_Retha: Command execution failed.: Process exited with an error: 1 (Exit value: 1)

To see the full stack trace of the errors, re-run Maven with the -e switch.
Re-run Maven using the -X switch to enable full debug logging.

For more information about the errors and possible solutions, please read the following articles:
[Help 1] http://wiki.apache.org/confluence/display/MAVEN/MojoExecutionException

```

### 2.2.2 Pertanyaan

1. Pada percobaan 2 diatas program yang dijalankan terjadi error, kemudian perbaiki sehingga program tersebut bisa dijalankan dan tidak error!

**Jawab :** Menambahkan method get pada ClassA dan Class B serta menambahkan extends pada ClassB terhadap ClassA

```

12 public class ClassA {
13     private int x;
14     private int y;
15
16     public void setX(int x) {
17         this.x = x;
18     }
19
20     public int getX() {
21         return x;
22     }
23
24     public void setY(int y) {
25         this.y = y;
26     }
27
28     public int getY() {
29         return y;
30     }
31
32     public void getNilai() {
33         System.out.println("nilai x: "+x);
34         System.out.println("nilai y: "+y);
35     }
36 }

```

```

11 public class ClassB extends ClassA{
12     private int z;
13
14     public void setZ(int z) {
15         this.z = z;
16     }
17
18     public int getZ() {
19         return z;
20     }
21
22     public void getNilaiZ() {
23         System.out.println("nilai Z: "+z);
24     }
25
26     public void getJumlah() {
27         System.out.println("jumlah : " + (getX()+getY()+z));
28     }
29 }

```

```

11 public class Percobaan2 {
12     public static void main(String[] args) {
13         ClassB hitung = new ClassB();
14         hitung.setX(20);
15         hitung.setY(30);
16         hitung.setZ(5);
17         hitung.getNilai();
18         hitung.getNilaiZ();
19         hitung.getJumlah();
20     }
21 }

```

**Hasilnya**

## INHERITENCE

```

-----< com.mycompany:Semester3_Retha >-----
] Building Semester3_Retha 1.0-SNAPSHOT
-----[ jar ]-----

] --- exec-maven-plugin:3.0.0:exec (default-cli) @ Semester3_Retha ---
nilai x: 20
nilai y: 30
nilai Z: 5
- jumlah : 55

-----
BUILD SUCCESS
-----

Total time: 1.878 s
Finished at: 2022-10-16T17:57:06+07:00
-----

```

2. Jelaskan apa penyebab program pada percobaan 1 ketika dijalankan terdapat error!

**Jawab :** Atribut class A dan class B dijadikan private sehingga hanya bisa diakses oleh class itu sendiri dan tidak dapat diturunkan.

### 2.3 Percobaan 3 – Super

#### 2.3.1 Langkah-Langkah Percobaan

1. Buatlah sebuah class parent/superclass dengan nama Bangun.java

```

12 public class Bangun {
13     protected double phi;
14     protected int r;
15 }

```

2. Buatlah sebuah class anak/subclass dengan nama Tabung.java

```

11 public class Tabung extends Bangun{
12     protected int t;
13     public void setSuperPhi(double phi){
14         super.phi = phi;
15     }
16     public void setSuperR(int r){
17         super.r = r;
18     }
19     public void setT(int t){
20         this.t = t;
21     }
22     public void volume(){
23         System.out.println("Volume Tabung adalah:"+(super.phi*super.r*super.r*this.t));
24     }
25 }

```



3. Buatlah class Percobaan3.java untuk menjalankan program diatas!

```

11      public class Percobaan3 {
12      public static void main(String[] args){
13          Tabung tabung = new Tabung();
14          tabung.setSuperPhi(3.14);
15          tabung.setSuperR(10);
16          tabung.setT(3);
17          tabung.volume();
18      }
19  }
20  
```

4. Jalankan program diatas!

```

-----< com.mycompany:Semester3_Retha >-----
[ Building Semester3_Retha 1.0-SNAPSHOT
  -----[ jar ]-----
  --- exec-maven-plugin:3.0.0:exec (default-cli) @ Semester3_Retha ---
  Volume Tabung adalah:942.0
  BUILD SUCCESS
  Total time: 3.217 s
  Finished at: 2022-10-16T18:09:13+07:00
  |
  
```

### 2.3.2 Pertanyaan

1. Jelaskan fungsi “super” pada potongan program berikut di class Tabung!

```

public void setSuperPhi(double phi){
    super.phi = phi;
}
public void setSuperR(int r){
    super.r = r;
}
  
```

**Jawab :** Untuk mengakses atribut pada class Bnagun yang merupakan parent dari class tabung. Atribut itu bersifat protected sehingga perlu kunci yaitu “super”

2. Jelaskan fungsi “super” dan “this” pada potongan program berikut di class Tabung!

```

public void volume(){
    System.out.println("Volume Tabung adalah: "+(super.phi*super.r*super.r*this.t));
}
  
```

**Jawab :** this untuk mengakses atribut dicass itu sendiri (class tabung) dan super untuk mengakses atribut class parent (class bangun).

## INHERITENCE

3. Jelaskan mengapa pada class Tabung tidak dideklarasikan atribut “phi” dan “r” tetapi class tersebut dapat mengakses atribut tersebut!

**Jawab :** Karena terdapat hak akses extends yang membolehkan tabung untuk bisa mengakses class bangun yang adalah parentnya.

### 2.4 Percobaan 4 – Super Constructor

#### 2.4.1 Langkah-Langkah Percobaan

1. Buatlah tiga file dengan nama ClassA.java , ClassB.java , dan ClassC.java, seperti pada kode program dibawah ini!

ClassA.java

```
12 public class ClassA {  
13     ClassA() {  
14         System.out.println("konstruktor A dijalankan");  
15     }  
16 }
```

ClassB.java

```
12 public class ClassB extends ClassA {  
13     ClassB() {  
14         System.out.println("konstruktor B dijalankan");  
15     }  
16 }
```

ClassC.java

```
11 public class ClassC extends ClassB {  
12     ClassC() {  
13         System.out.println("konstruktor C dijalankan");  
14     }  
15 }
```

2. Buatlah class Percobaan4.java untuk menjalankan program diatas!

```
11 public class Percobaan4 {  
12     public static void main(String[] args) {  
13         ClassC test = new ClassC();  
14     }  
15 }
```

3. Jalankan program kemudian amati apa yang terjadi!

```

-----< com.mycompany:Semester3_Retha >-----
[ Building Semester3_Retha 1.0-SNAPSHOT
-----[ jar ]-----
[ --- exec-maven-plugin:3.0.0:exec (default-cli) @ Semester3_Retha ---
    konstruktor A dijalankan
    konstruktor B dijalankan
    konstruktor C dijalankan
-----
BUILD SUCCESS
-----
Total time:  2.877 s
Finished at: 2022-10-16T18:18:12+07:00
-----

```

## 2.4.2 Pertanyaan

1. Pada percobaan 4 sebutkan mana class yang termasuk superclass dan subclass, kemudian jelaskan alasannya!

**Jawab :** ClassA termasuk superclass karena mempunyai class turunan yaitu ClassB dan ClassC

ClassB dan ClassC termasuk subclass karena menjadi class turunan dari ClassA dan mewarisi atribut yang terdapat pada ClassA

2. Ubahlah isi konstruktor default ClassC seperti berikut:

```

public class ClassC extends ClassB{
    ClassC(){
        super();
        System.out.println("konstruktor C dijalankan");
    }
}

```

Tambahkan kata super() di baris Pertama dalam konstruktor defaultnya. Coba jalankan kembali class Percobaan4 dan terlihat tidak ada perbedaan dari hasil outputnya!

**Jawab :**

```

public class ClassC extends ClassB{
    ClassC(){
        super();
        System.out.println("konstruktor C dijalankan");
    }
}

```

```

[ --- exec-maven-plugin:3.0.0:exec
    konstruktor A dijalankan
    konstruktor B dijalankan
    konstruktor C dijalankan
-----
BUILD SUCCESS
-----
Total time:  2.840 s

```

3. Ublah isi konstruktor default ClassC seperti berikut:

```

12 public class ClassC extends ClassB{
13     ClassC() {
14         System.out.println("konstruktor C dijalankan");
15         super();
16     }
17 }

```

Ketika mengubah posisi `super()` dibaris kedua dalam konstruktor defaultnya dan terlihat ada error. Kemudian kembalikan `super()` kebaris pertama seperti sebelumnya, maka errornya akan hilang.

Perhatikan hasil keluaran ketika class Percobaan4 dijalankan. Kenapa bisa tampil output seperti berikut pada saat instansiasi objek test dari class ClassC

```

Output - Percobaan4 (run)

run:
konstruktor A dijalankan
konstruktor B dijalankan
konstruktor C dijalankan
BUILD SUCCESSFUL (total time: 0 seconds)

```

Jelaskan bagaimana urutan proses jalannya konstruktor saat objek test dibuat!

**Jawab :** ClassA sebagai parent dari ClassB dan ClassC sebagai subclass. Akses ClassA lebih dulu, kemudian mengakses ClassB dan ClassC

4. Apakah fungsi `super()` pada potongan program dibawah ini di ClassC!

```

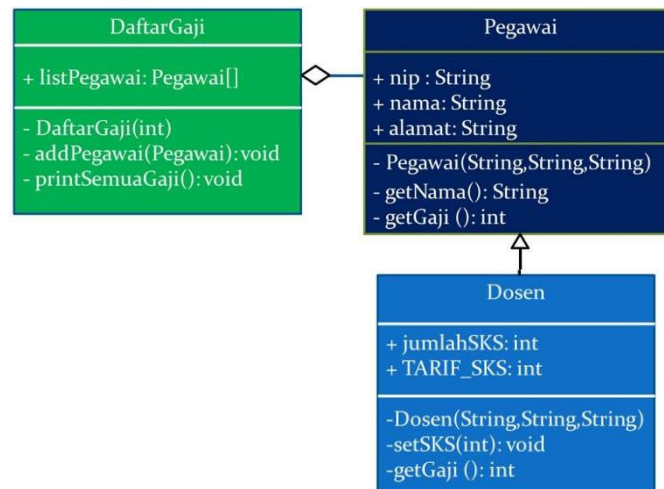
public class ClassC extends ClassB{
    ClassC() {
        super();
        System.out.println("konstruktor C dijalankan");
    }
}

```

**Jawab :** Untuk mengakses konstruktor parents yang hanya bisa diakses jika letaknya terdapat di posisi pertama setelah konstruktor subclass

## TUGAS

Buatlah sebuah program dengan konsep pewarisan seperti pada class diagram berikut ini. Kemudian buatlah instansiasi objek untuk menampilkan data nama pegawai dan gaji yangdidapatkannya.



**Jawab :**

Class Pegawai

```

public class Pegawai {
    public String nip;
    public String nama;
    public String alamat;
    public int gaji;

    Pegawai(String nip, String nama, String alamat){
        this.nip = nip;
        this.nama = nama;
        this.alamat = alamat;
    }

    public String getNama(){
        return nama;
    }

    public int getGaji(){
        return gaji;
    }
}
    
```

Class Dosen

## INHERITENCE

```

11 public class Dosen extends Pegawai{
12     public int jumlahSKS;
13     public int tarif_SKS;
14
15     public Dosen(String nip, String nama, String alamat, int jumlahSKS, int tarif_SKS){
16         super(nip, nama, alamat);
17         this.jumlahSKS = jumlahSKS;
18         this.tarif_SKS = tarif_SKS;
19     }
20     public void setSKS(int sks){
21         this.jumlahSKS = jumlahSKS;
22     }
23     public int getGaji(){
24         int gaji = (jumlahSKS * tarif_SKS);
25         return gaji;
26     }
27 }

```

### Class DaftarGaji

```

public class DaftarGaji {
    public Pegawai[] listPegawai;
    public int jumlah;
    public Dosen[] dosen;

    DaftarGaji(int jumlah){
        this.listPegawai = new Pegawai[jumlah];
        this.dosen = new Dosen[jumlah];
    }

    public void addPegawai(Pegawai pegawai, Dosen dosen){
        this.listPegawai[jumlah] = pegawai;
        this.dosen[jumlah] = dosen;
        jumlah++;
    }

    public void printSemuaGaji(){
        for(int i = 0; i < jumlah; i++){
            System.out.println("CETAK GAJI PEGAWAI");
            System.out.println("NIP      : "+listPegawai[i].nip);
            System.out.println("Nama    : "+listPegawai[i].nama);
            System.out.println("Alamat  : "+listPegawai[i].alamat);
            System.out.println("-----");
            System.out.println("Total gaji : "+dosen[i].getGaji());
            System.out.println();
        }
    }
}

```

### Class Main

## INHERITENCE

```
public class MainTugas {  
    public static void main(String[] args) {  
        DaftarGaji dg = new DaftarGaji(2);  
        Dosen d1 = new Dosen("2131762056", "Marcella A.", "Malang", 8, 300000);  
        Dosen d2 = new Dosen("2141762057", "Rudyantoro Siswa", "Surabaya", 5, 400000);  
        dg.addPegawai(d1, d1);  
        dg.addPegawai(d2, d2);  
        dg.printSemuaGaji();  
    }  
}
```

Output :

```
--- exec-maven-plugin:3.0.0:exec (o  
CETAK GAJI PEGAWAI  
NIP      : 2131762056  
Nama     : Marcella A.  
Alamat   : Malang  
-----  
Total gaji : 2400000  
  
CETAK GAJI PEGAWAI  
NIP      : 2141762057  
Nama     : Rudyantoro Siswa  
Alamat   : Surabaya  
-----  
Total gaji : 2000000  
  
-----  
BUILD SUCCESS  
-----
```