

# Tugas Praktikum 6

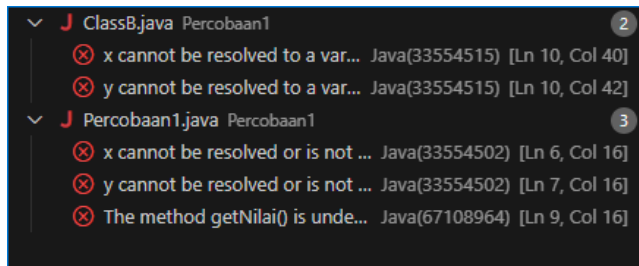
## Inheritance 2



Rafi Ody Prasetyo  
(2341720180)

D-IV Teknik Informatika  
Politeknik Negeri Malang  
Semester 3  
2024

## Percobaan 1



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

### Jawab:

Agar program tidak terjadi error, pada classB dapat ditambahkan extends classA agar variabel yang terdapat pada classA terbaca.

```
package Percobaan1;

public class ClassB extends ClassA {

    public int z;

    public void getNilaiZ() {
        System.out.println("nilai z:" + z);
    }
    public void getJumlah() {
        System.out.println("Jumlah:" + (x+y+z));
    }

}
```

Output:

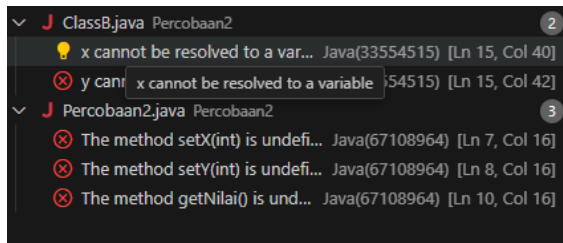
```
nilai x:20
nilai y:30
nilai z:5
Jumlah:55
```

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

### Jawab:

Karena pada classB belum terhubung dengan classA, sehingga variabel x dan y yang terdapat di dalam classA tidak dapat dipanggil. Untuk menghubungkannya kita hanya perlu menambahkan extends classA setelah deklarasi classB.

## Percobaan 2



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

### Jawab:

Error yang terjadi pada percobaan 2 sama dengan error pada percobaan 1. Untuk mengatasinya, pada deklarasi classB hanya perlu ditambahkan extends classA untuk menjangkau class parent.

```
package Percobaan2;

public class ClassB extends ClassA{

    public int z;

    public void setZ(int z) {
        this.z = z;
    }

    public void getNilaiZ() {
        System.out.println("nilai z:" + z);
    }
    public void getJumlah() {
        System.out.println("Jumlah:" + (x+y+z));
    }

}
```

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

### Jawab:

```
nilai x:20
nilai y:30
nilai z:5
Jumlah:55
```

Karena pada classB belum terhubung dengan classA, sehingga variabel x dan y yang terdapat di dalam classA tidak dapat dipanggil,

### Percobaan 3

```
de_ws (30350000_1703001)@bin: ~$ java -cp . Percobaan3
Volume Tabung adalah: 942.0
```

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 yang berada pada class parent.

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:

Super digunakan untuk mengakses atribut yang berada pada kelas lain atau kelas parent, sedangkan this digunakan untuk mengakses atribut di kelas yang sama.

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

#### Jawab:

Karena class Tabung sudah terhubung dengan class Bangun yang mempunyai atribut phi dan r. Sehingga class Tabung dapat mengakses atribut yang berada pada class Bangun.

## Percobaan 4

```
Konstruktor A dijalankan
Konstruktor B dijalankan
Konstruktor C dijalankan
```

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

**Jawab:**

ClassA merupakan superclass karena tidak ada class yang mewarisi. ClassB merupakan subclass karena kelas tersebut diwarisi oleh ClassA. ClassC merupakan subclass karena kelas tersebut diwarisi ClassB, dan secara tidak langsung diwarisi juga oleh ClassA.

2. Ubahlah 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:**

```
Konstruktor A dijalankan
Konstruktor B dijalankan
Konstruktor C dijalankan
```

3. Ubahlah 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() di baris kedua dalam konstruktor defaultnya dan terlihat ada error. Kemudian kembalikan super() ke baris 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:**

Konstruktor `super()` harus berada pada baris pertama statement konstruktor yang berada pada subclass. Urutan output diatas, dimulai dari superclass yaitu ClassA. Kemudian akan diikuti kelas yang diwarisi ClassA yaitu ClassB. Dan yang terakhir adalah ClassC yang diwarisi oleh ClassB.

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

```

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

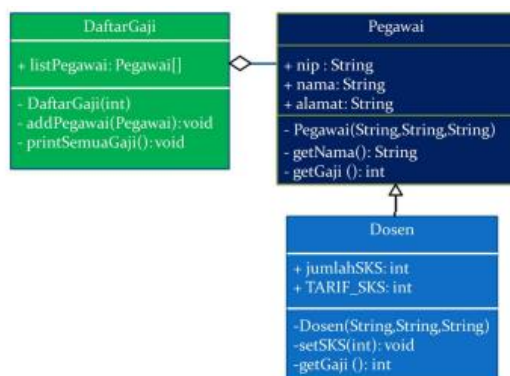
```

**Jawab:**

Fungsi `super()` pada potongan program di ClassC bertujuan untuk memanggil constructor dari kelas induknya, yaitu ClassB, sebelum melanjutkan eksekusi kode pada constructor ClassC. Dengan memanggil `super()`, kode memastikan bahwa inisialisasi yang dilakukan di ClassB juga terjadi ketika objek dari ClassC dibuat.

## Tugas

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



## Pegawai.java

Disini class Pegawai.java berperan sebagai superclass karena mewarisi class Dosen.java.

```
package Tugas;

public class Pegawai {

    private String nip;
    private String nama;
    private String alamat;

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

    public String getNama() {
        return this.nama;
    }

    public int getGaji() {
        return 0;
    }

}
```

## Dosen.java

Class Dosen.java dijadikan sebagai subclass, karena class ini diwarisi oleh class Pegawai.java

```
package Tugas;

public class Dosen extends Pegawai {

    private int jumlahSKS;
    private int tarifSKS = 50000;

    public Dosen(String nip, String nama, String alamat) {
        super(nip, nama, alamat);
    }

    public void setSKS(int jumlahSKS) {
        this.jumlahSKS = jumlahSKS;
    }

    public int getGaji() {
        return jumlahSKS * tarifSKS;
    }

}
```

## DaftarGaji.java

Class ini memiliki relas dengan class Pegawai.java, yang digunakan untuk mengambil data nama dan gaji. Data tersebut akan dimasukkan kedalam attribut array listPegawai yang bertipe data class Pegawai.

```
package Tugas;

public class DaftarGaji {

    private Pegawai[] listPegawai;

    public DaftarGaji(int jumlahPegawai) {
        this.listPegawai = new Pegawai[jumlahPegawai];
    }

    public void addPegawai(Pegawai pegawai) {
        for (int i = 0; i < listPegawai.length; i++) {
            if (listPegawai[i] == null) {
                listPegawai[i] = pegawai;
                break;
            }
        }
    }

    public void printSemuaGaji() {
        for (Pegawai pegawai : listPegawai) {
            if (pegawai != null) {
                System.out.println("Nama: " + pegawai.getNama() +
                    " | Gaji: " + pegawai.getGaji());
            }
        }
    }
}
```



## MainPegawai.java

```
package Tugas;

public class MainPegawai {

    public static void main(String[] args) {

        DaftarGaji daftarGaji = new DaftarGaji(2);

        Pegawai dosen1 = new Dosen("N12", "Ilham", "Malang");
        ((Dosen) dosen1).setSKS(12);
        Pegawai dosen2 = new Dosen("N18", "Yusuf", "Malang");
        ((Dosen) dosen2).setSKS(10);

        daftarGaji.addPegawai(dosen1);
        daftarGaji.addPegawai(dosen2);

        daftarGaji.printSemuaGaji();

    }

}
```

## Output:

```
Nama: Ilham | Gaji: 600000
Nama: Yusuf | Gaji: 500000
```