

Object Oriented Programming Inheritance

**Name**

Virza Aulia Rachman

NIM

2241720078

Class

1i

Department

Information Technology

Study Program

D4 Informatics Engineering

Practicum 1

```
package Practicum1;

no usages
public class ClassA {
    1 usage  2 related problems
    public int x;
    1 usage  2 related problems
    public int y;

    no usages
    public void getNilai(){
        System.out.println("nilai x: " +x);
        System.out.println("nilai y: " +y);
    }
}
```

```
package Practicum1;

2 usages
public class ClassB {
    3 usages
    public int z;
    1 usage
    public void getNilaiZ(){
        System.out.println("nilai Z :"+z );
    }
    1 usage
    public void getJumlah(){
        System.out.println( x: "Jumlah:"+(x+y+z));
    }
}
```

```
package Practicum1;

public class Percobaan1 {
    public static void main(String[]args){
        ClassB hitung = new ClassB();
        hitung.x = 20;
        hitung.y = 30;
        hitung.z = 5;
        hitung.getNilai();
        hitung.getNilaiZ();
        hitung.getJumlah();
    }
}
```

Question

1. Pada percobaan 1 diatas program yang dijalankan terjadi error, kemudian perbaiki sehingga kemudian perbaiki sehingga program tersebut dapat dijalankan dan tidak error
2. Jelaskan apa penyebab program pada percobaan 1 ketika dijalankan terdapat error!

Answer :

1. ClassB and Percobaan1

```
package Practicum1;

2 usages
public class ClassB {
3 usages
    public int z;

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

1 usage
    public void getJumlah(ClassA objA) {
        System.out.println("Jumlah: " + (objA.x + objA.y + z));
    }
}
```

```
package Practicum1;

public class Percobaan1 {
    public static void main(String[] args) {
        ClassA objA = new ClassA();
        objA.x = 20;
        objA.y = 30;

        ClassB hitung = new ClassB();
        hitung.z = 5;

        objA.getNilai();
        hitung.getNilaiZ();
        hitung.getJumlah(objA);
    }
}
```

2. Karena ClassB belum dihubungkan dengan ClassA sehingga terjadi error pad atribut objek.

Practicum 2

```
package Practicum2;

1 usage 1 inheritor
public class ClassA {
    3 usages
    protected int x;
    3 usages
    protected int y;
    1 usage
    public void setX(int x) {
        this.x = x;
    }
    1 usage
    public void setY(int y) {
        this.y = y;
    }
    1 usage
    public void getNilai() {
        System.out.println("Nilai x: " + x);
        System.out.println("Nilai y: " + y);
    }
}
```

```
package Practicum2;

2 usages
public class ClassB extends ClassA {
    3 usages
    protected int z;
    1 usage
    public void setZ(int z) {
        this.z = z;
    }
    1 usage
    public void getNilaiZ() {
        System.out.println("Nilai Z: " + z);
    }
    1 usage
    public void getJumlah() {
        System.out.println("Jumlah: " + (x + y + z));
    }
}
```

```
package Practicum2;

public class Percobaan2 {
    public static void main(String[] args) {
        ClassB hitung = new ClassB();
        hitung.setX(20);
        hitung.setY(30);
        hitung.setZ(5);

        hitung.getNilai();
        hitung.getNilaiZ();
        hitung.getJumlah();
    }
}
```

Question

1. Pada percobaan 1 diatas program yang dijalankan terjadi error, kemudian perbaiki sehingga kemudian perbaiki sehingga program tersebut dapat dijalankan dan tidak error
2. Jelaskan apa penyebab program pada percobaan 1 ketika dijalankan terdapat error!

Answer:

1. ClassA,ClassB,Percobaan2

```
package Practicum2;
1 usage 1 inheritor
public class ClassA {
    3 usages
    protected int x;
    3 usages
    protected int y;
    1 usage
    public void setX(int x) {
        this.x = x;
    }
    1 usage
    public void setY(int y) {
        this.y = y;
    }
    1 usage
    public void getNilai() {
        System.out.println("Nilai x: " + x);
        System.out.println("Nilai y: " + y);
    }
}
```

```
package Practicum2;

2 usages
public class ClassB extends ClassA {
    3 usages
    protected int z;
    1 usage
    public void setZ(int z) {
        this.z = z;
    }
    1 usage
    public void getNilaiZ() {
        System.out.println("Nilai Z: " + z);
    }
    1 usage
    public void getJumlah() {
        System.out.println("Jumlah: " + (x + y + z));
    }
}
```

```
package Practicum2;

public class Percobaan2 {
    public static void main(String[] args) {
        ClassB hitung = new ClassB();
        hitung.setX(20);
        hitung.setY(30);
        hitung.setZ(5);

        hitung.getNilai();
        hitung.getNilaiZ();
        hitung.getJumlah();
    }
}
```

2. Karena inheritance tidak diimplementasikan dengan benar untuk mengakses variabel dan metode dari kelas lain.

Practicum 3

```
package Practicum3;

1 usage 1 inheritor
public class Bangun {
    2 usages
    protected double phi;
    3 usages
    protected int r;
}
```

```
package Practicum3;

public class percobaan3 {
    public static void main(String[] args){
        Tabung tabung = new Tabung();
        tabung.setSuperPhi(3.14);
        tabung.setSuperR(10);
        tabung.setT(3);
        tabung.Volume();
    }
}
```

```
package Practicum3;

2 usages
public class Tabung extends Bangun{
    2 usages
    protected int t;
    1 usage
    public void setSuperPhi(Double phi){
        super.phi = phi;
    }
    1 usage
    public void setSuperR(int r){
        super.r = r;
    }
    1 usage
    public void setT(int t){
        this.t = t;
    }

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

Question

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;  
}
```

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.h));  
}
```

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

Answer:

1. Kata kunci "super" digunakan dalam class Tabung untuk merujuk dan mengakses variabel phi dan r yang merupakan anggota dari class induk (superclass) Bangun.
2. super digunakan untuk mengakses variabel dan metode yang diwarisi dari kelas induk, sementara this digunakan untuk merujuk pada variabel dan metode yang ada dalam kelas saat ini.
3. Class Tabung dapat mengakses atribut "phi" dan "r" karena mereka diwarisi dari class Bangun yang menjadi superclass-nya.

Practicum 4

```
package Practicum4;

1 usage 2 inheritors
public class ClassA {
    1 usage
    ClassA(){
        System.out.println("Konstruktor A dijalankan");
    }
}
```

```
package Practicum4;

1 usage 1 inheritor
public class ClassB extends ClassA{
    1 usage
    ClassB(){
        System.out.println("konstruktor B dijalankan");
    }
}
```

```
package Practicum4;

2 usages
public class ClassC extends ClassB{
    1 usage
    ClassC(){
        System.out.println("konstruktor c dijalankan");
    }
}
```

```
package Practicum4;

public class percobaan4 {
    public static void main(String[] args){
        ClassC test = new ClassC();
    }
}
```


Question

1. Pada percobaan 4 sebutkan mana class yang termasuk superclass dan subclass, kemudian jelaskan alasannya!
2. Ubahlah isi konstruktor default ClassC seperti berikut:

```
public class ClassC extends ClassB{  
    ClassC(){  
        super();  
        System.out.println("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()` 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!

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

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

Answer

1. ClassA adalah superclass, sedangkan ClassB dan ClassC adalah subclass karena ClassB dan ClassC mengextends (mewarisi) dari ClassA, sehingga ClassA menjadi kelas induk yang mewariskan konstruktor dan properti kepada kedua subclass tersebut.
2. This is the result

```
Konstruktor A dijalankan  
konstruktor B dijalankan  
konstruktor c dijalankan
```

3. Konstruktor ClassA dijalankan terlebih dahulu karena ClassC memanggil super(), kemudian konstruktor ClassB dijalankan, dan terakhir, konstruktor ClassC dijalankan.
4. Pemanggilan super(); dalam ClassC digunakan untuk menjalankan konstruktor dari kelas induknya, yaitu ClassB, sebelum menjalankan konstruktor ClassC itu sendiri.

Assignment

Pegawai

```
package Assignment;

4 usages 1 inheritor
public class Pegawai {
    1 usage
    private String nip;
    2 usages
    private String nama;
    1 usage
    private String alamat;
    no usages
    private int gaji;
    1 usage
    Pegawai(String nip,String nama,String alamat){
        this.nip = nip;
        this.nama = nama;
        this.alamat = alamat;
    }

    1 usage
    public String getNama() {
        return nama;
    }

    1 usage 1 override
    public int getGaji() {
        return 0;
    }
}
```

Dosen

```
package Assignment;

2 usages
public class Dosen extends Pegawai {
    2 usages
    private int JumlahSKS;
    2 usages
    private int TarifSKS;

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

    1 usage
    public void setJumlahSKS(int jumlahSKS) {
        JumlahSKS = jumlahSKS;
    }

    1 usage
    public void setTarifSKS(int tarifSKS) {
        TarifSKS = tarifSKS;
    }

    1 usage
    public int getGaji() {
        return TarifSKS*JumlahSKS;
    }
}
```

DaftarGaji

```
package Assignment;

2 usages
public class Daftargaji {
    8 usages
    Pegawai [] listpegawai;

    1 usage
    public Daftargaji(int jumlahPegawai){
        listpegawai = new Pegawai[jumlahPegawai];
    }

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

    1 usage
    public void PrintSemuaGaji(){
        for (int i = 0;i< listpegawai.length;i++){
            if(listpegawai[i] != null){
                System.out.println("Nama dan gaji: " + listpegawai[i].getNama() + " " +listpegawai[i].getGaji());
            }
        }
    }
}
```

Main and result

```
package Assignment;

public class Main {
    public static void main(String[] args) {
        Daftargaji Polinema = new Daftargaji( jumlahPegawai: 1);
        Dosen woilah = new Dosen( nip: "8787", nama: "Virza Goks", alamat: "Kembang Kertas");
        woilah.setJumlahSKS(10);
        woilah.setTarifSKS(1000000);
        Polinema.addPegawai(woilah);
        Polinema.PrintSemuaGaji();
    }
}
```

Nama dan gaji: Virza Goks 1000000

Process finished with exit code 0