

OBJECT ORIENTED PROGRAMMING



By :

Sri Kresna Maha Dewa

2241720244

STUDY PROGRAM D-IV INFORMATIC ENGINEERING

INFORMATION TECHNOLOGY DEPARTMENT

MALANG STATE POLYTECHNIC

Soekarno Hatta Street No.9, Jatimulyo, Lowokwaru District, Malang City, East Java

65141

PERCOBAAN 1

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

```
3 public class ClassB extends ClassA{
4     public int z;
5
6     public void getNilaiZ() {
7         System.out.println("nilai Z : " + z);
8     }
9
10    public void getJumlah() {
11        System.out.println("jumlah : " + (x + y + z));
12    }
13 }
14
```

```
nilai x : 20
nilai y : 30
nilai Z : 5
jumlah : 55
PS D:\remote\code-for-edu>
```

2. Jelaskan apa penyebab program pada percobaan 1 ketika dijalankan terdapat error!
Error terjadi karena classB ingin mengakses attribute x dan y tetapi dia tidak melakukan inheritance terlebih dahulu.

PERCOBAAN 2

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

```
3 public class ClassA {
4     private int x;
5     private int y;
6
7     public void setX(int x) {
8         this.x = x;
9     }
10
11    public void setY(int y){
12        this.y = y;
13    }
14
15    public int getX() {
16        return x;
17    }
18
19    public int getY(){
20        return y;
21    }
22
23    public void getNilai() {
24        System.out.println("nilai x : " + x);
25        System.out.println("nilai y : " + y);
26    }
27 }
28
```

```
3 public class ClassB extends ClassA {
4     private int z;
5
6     public void setZ(int z) {
7         this.z = z;
8     }
9
10    public void getNilaiZ() {
11        System.out.println("nilai Z : " + z);
12    }
13
14    public void getJumlah() {
15        System.out.println("jumlah : " + (getX() + getY() + z));
16    }
17 }
18
```

```
nilai x : 20
nilai y : 30
nilai Z : 5
jumlah : 55
PS D:\remote\code-for-edu>
```

2. Jelaskan apa penyebab program pada percobaan 1 ketika dijalankan terdapat error! Error terjadi karena classB ingin mengakses attribute x dan y tetapi attribute tersebut telah di encapsulation sehingga tidak dapat diakses secara langsung. Jadi kita perlu mendefinisikan getter-setter method agar attribute dapat diakses.

PERCOBAAN 3

1. Jelaskan fungsi “super” pada potongan program berikut di class Tabung! Fungsi super pada potongan program tersebut berfungsi untuk mengakses attribute dari superclass/parent.
2. Jelaskan fungsi “super” dan “this” pada potongan program berikut di class Tabung! Pada potongan program tersebut kita ingin menghitung volume dari tabung, tetapi kita butuh attribute phi dan r yang perlu diambil dari parent class. Oleh karena itu, kita memakai fungsi super, selanjutnya this digunakan untuk mendapatkan attribute t dari class tabung.
3. Jelaskan mengapa pada class Tabung tidak dideklarasikan attribute “phi” dan “r” tetapi class tersebut dapat mengakses atribut tersebut. Hal ini karena class tabung adalah child class/ subclass dari Bangun. Hal ini terjadi karena sebuah child class akan mewarisi semua method dan attribute dari parentclass.

PERCOBAAN 4

1. Pada percobaan 4 sebutkan mana class yang termasuk superclass dan subclass, kemudian jelaskan alasannya! Class A adalah superclass dari Class B, Class B adalah superclass dari Class C. Sedangkan Class B adalah subclass dari Class A, dan Class C adalah subclass dari Class B.
2. Ubahlah isi konstruktor default ClassC seperti berikut.

```

1  package oop_java.week7.percobaan4;
2
3  public class ClassC extends ClassB{
4      ClassC() {
5          super();
6          System.out.println(x:"konstruktor C dijalankan");
7      }
8  }
9

```

3. Jelaskan bagaimana urutan proses jalannya konstruktor saat objek test dibuat!
Kontruktor classC akan dipanggil, karena classC adalah subclass dari classB maka konstruktor classB juga akan terpanggil. Begitu juga dengan classA karena classB adalah subclass dari ClassA.
4. Apa fungsi super() pada potongan program dibawah ini di classC
Fungsi super() dapat digunakan untuk memanggil konstruktor dari superclass/parentclass

TUGAS

```

1  package oop_java.week7.Tugas;
2
3  public class Pegawai {
4      private String nip;
5      private String nama;
6      private String alamat;
7      private int gaji;
8
9      public Pegawai(String nip, String nama, String alamat) {
10         this.nip = nip;
11         this.nama = nama;
12         this.alamat = alamat;
13     }
14
15     public String getNama() {
16         return nama;
17     }
18
19     public int getGaji() {
20         return gaji;
21     }
22
23
24 }
25

```

```
1 package oop_java.week7.Tugas;
2
3 public class Dosen extends Pegawai{
4     private int jumlahSKS;
5     private int TARIF_SKS;
6
7     public Dosen(String nip, String nama, String alamat) {
8         super(nip, nama, alamat);
9         TARIF_SKS = 100000;
10    }
11
12    public void setSKS(int jumlahSKS) {
13        this.jumlahSKS = jumlahSKS;
14    }
15
16    @Override
17    public int getGaji() {
18        return jumlahSKS * TARIF_SKS;
19    }
20
21 }
22
```

```
1 package oop_java.week7.Tugas;
2
3 public class DaftarGaji{
4     Pegawai[] listPegawai;
5
6     public DaftarGaji(int jumlahPegawai) {
7         listPegawai = new Pegawai[jumlahPegawai];
8     }
9
10    public void addPegawai(Pegawai pegawai) {
11        for (int i = 0; i < listPegawai.length; i++) {
12            if (listPegawai[i] == null) {
13                listPegawai[i] = pegawai;
14                break;
15            }
16        }
17    }
18
19    public void printSemuaGaji() {
20        for (int i = 0; i < listPegawai.length; i++) {
21            if (listPegawai[i] != null) {
22                System.out.println("Nama: " + listPegawai[i].getNama());
23                System.out.println("Gaji: " + listPegawai[i].getGaji());
24                System.out.println();
25            }
26        }
27    }
28 }
29
```

```
1 package oop_java.week7.Tugas;
2
3 /**
4  * Main
5  */
6 public class Main {
7     public static void main(String[] args) {
8         DaftarGaji daftarGaji = new DaftarGaji(10);
9         Dosen dosen1 = new Dosen("123", "Dosen 1", "Alamat 1");
10        Dosen dosen2 = new Dosen("456", "Dosen 2", "Alamat 2");
11        Dosen dosen3 = new Dosen("789", "Dosen 3", "Alamat 3");
12        dosen1.setSKS(10);
13        dosen2.setSKS(20);
14        dosen3.setSKS(30);
15        daftarGaji.addPegawai(dosen1);
16        daftarGaji.addPegawai(dosen2);
17        daftarGaji.addPegawai(dosen3);
18        daftarGaji.printSemuaGaji();
19    }
20 }
```

```
Nama: Dosen 1
Gaji: 1000000

Nama: Dosen 2
Gaji: 2000000

Nama: Dosen 3
Gaji: 3000000

PS D:\remote\code-for-edu>
```