

Nama : Welson Mario Naibaho

Kelas : TI 2C

Mata Kuliah : Praktikum PBO

LAPORAN PERCOBAAN

1. AbstractClass

Class Hewan

```
13 public abstract class Hewan {
14     private int umur;
15
16     protected Hewan() {
17         this.umur=0;
18     }
19
20     public void bertambahUmur() {
21         this.umur += 1;
22     }
23
24     public void bergerak() {
25
26     }
27 }
```

Class Ikan

```
11 public class Ikan extends Hewan{
12     @Override
13     public void bergerak() {
14         System.out.println("Berenang dengan SIRIP, \"wush..wush..\"");
15     }
16 }
```

Class Kucing

```
11 public class Kucing extends Hewan{
12     public void bergerak() {
13         System.out.println("Berjalan dengan KAKI, \"Tap..tap..\"");
14     }
15 }
```

Class Orang

```
11 public class Orang {
12
13     private String nama;
14     private Hewan hewanPeliharaan;
15
16     Orang(String nama) {
17         this.nama = nama;
18     }
19
20     public void peliharaHewan(Hewan peliharaan) {
21         this.hewanPeliharaan = peliharaan;
22     }
23
24     public void ajakPeliharaanJalan() {
25         System.out.println("Nama : " + nama);
26         System.out.println("Hewan Peliharaanku Bergerak dengan : ");
27         this.hewanPeliharaan.bergerak();
28         System.out.println("-----");
29     }
30 }
```

Class Program (Main)

```
11 public class Program {
12
13     public static void main(String[] args) {
14         Kucing kucingKampung = new Kucing();
15         Ikan lumba = new Ikan();
16
17         Orang ani = new Orang("Ani");
18         Orang budi = new Orang("Budi");
19
20         ani.peliharaHewan(kucingKampung);
21         budi.peliharaHewan(lumba);
22
23         ani.ajakPeliharaanJalan();
24         budi.ajakPeliharaanJalan();
25     }
26 }
```

Output

```
Nama : Ani
Hewan Peliharaanku Bergerak dengan :
Berjalan dengan KAKI, "Tap..tap.."
-----
Nama : Budi
Hewan Peliharaanku Bergerak dengan :
Berenang dengan SIRIP, "wush..wush.."
-----
BUILD SUCCESSFUL (total time: 0 seconds)
```

Pertanyaan

1. Bolehkah apabila sebuah class yang meng-extends suatu abstract class tidak mengimplementasikan method abstract yang ada di class induknya?Buktikan

Jawaban : Tidak boleh, contoh pada class kucing, apabila tidak mengimplementasikan method bergerak yang didapat dari parent, maka program akan error

2. InterfaceClass

Class ICumlaude

```
14 public interface ICumlaude {  
    public abstract void lulus();  
    public abstract void meraihIPKTinggi();  
}
```

Class Mahasiswa

```
12 public class Mahasiswa {  
13     protected String nama;  
14     public Mahasiswa(String nama){  
15         this.nama=nama;  
16     }  
17     public void kuliahDiKampus(){  
18         System.out.println("Aku mahasiswa, namaku " + this.nama);  
19         System.out.println("Aku berkuliah di kampus.");  
20     }  
21 }
```

Class Sarjana

```
11 public class Sarjana extends Mahasiswa implements ICumlaude {  
12  
13     public Sarjana(String nama){  
14         super(nama);  
15     }  
16  
17     @Override  
18     public void lulus() {  
19         System.out.println("Nama Saya : "+nama);  
20         System.out.println("Aku sudah menyelesaikan SKRIPSI");  
21     }  
22  
23     @Override  
24     public void meraihIPKTinggi() {  
25         System.out.println("IPK-Ku lebih dari 3,51");  
26     }  
27  
28 }
```

Class PascaSarjana

```
public class PascaSarjana extends Mahasiswa implements ICumlaude, IBerprestasi {

    public PascaSarjana(String nama){
        super(nama);
    }

    @Override
    public void lulus() {
        System.out.println("Nama Saya: " + nama);
        System.out.println("Aku sudah menyelesaikan TESIS");
    }

    @Override
    public void meraihIPKTinggi() {
        System.out.println("IPK-Ku lebih dari 3,71");
    }

    @Override
    public void menjuaraiKompetisi() {
        System.out.println("Saya telah menjuarai Kompetisi INTERNASIONAL");
    }

    @Override
    public void membuatPublikasiIlmiah() {
        System.out.println("Saya menerbitkan artikel di jurnal INTERNASIONAL");
    }

}
```

Class Rektor

```
public class Rektor {
    public void beriSertifikatCumlaude(ICumlaude mahasiswa){
        System.out.println("Saya REKTOR, memberikan sertifikat cumlaude");
        System.out.println("Selamat! silahkan perkenalkan diri Anda..");

        mahasiswa.lulus();
        mahasiswa.meraihIPKTinggi();

        System.out.println("-----");
    }
    public void beriSertifikatMawapres(IBerprestasi mahasiswa){
        System.out.println("Saya REKTOR, memberikan sertifikat MAWAPRES");
        System.out.println("Selamat! Bagaimana Anda bisa berprestasi?");

        mahasiswa.menjuaraiKompetisi();
        mahasiswa.membuatPublikasiIlmiah();

        System.out.println("-----");
    }
}
```

Class Interfacemain

```
5 package pbominggul0.interfaceclass;
6 import pbominggul0.interfaceclass.ICumlaude;
7 import pbominggul0.interfaceclass.Mahasiswa;
8 import pbominggul0.interfaceclass.PascaSarjana;
9 import pbominggul0.interfaceclass.Rektor;
10 import pbominggul0.interfaceclass.Sarjana;
11 /**
12  *
13  * @author lenovo
14  */
15 public class interfacemain {
16     public static void main(String[] args) {
17
18         Rektor pakrektor = new Rektor();
19
20         Mahasiswa mhs = new Mahasiswa("Charlie");
21         Sarjana scl = new Sarjana("Dini");
22         PascaSarjana pscl = new PascaSarjana("Elok");
23
24         //pakrektor.beriSertifikatCumlaude(msh);
25         pakrektor.beriSertifikatCumlaude(scl);
26         pakrektor.beriSertifikatCumlaude(pscl);
27
28         //pakrektor.beriSertifikatMawapres(scl);
29         pakrektor.beriSertifikatMawapres(pscl);
30
31     }
32 }
```

Output

```
Saya REKTOR, memberikan sertifikat cumlaude
Selamat! silahkan perkenalkan diri Anda..
Nama Saya : Dini
Aku sudah menyelesaikan SKRIPSI
IPK-Ku lebih dari 3,51
-----
Saya REKTOR, memberikan sertifikat cumlaude
Selamat! silahkan perkenalkan diri Anda..
Nama Saya: Elok
Aku sudah menyelesaikan TESIS
IPK-Ku lebih dari 3,71
-----
Saya REKTOR, memberikan sertifikat MAWAPRES
Selamat! Bagaimana Anda bisa berprestasi?
Saya telah menjuarai Kompetisi INTERNASIONAL
Saya menerbitkan artikel di jurnal INTERNASIONAL
-----
BUILD SUCCESSFUL (total time: 0 seconds)
```

TUGAS

Class Binatang

```
12 public abstract class Binatang {
13     protected String nama;
14     protected int jmlKaki;
15
16     public Binatang(String nama, int jmlKaki) {
17         this.nama = nama;
18         this.jmlKaki = jmlKaki;
19     }
20
21     public void setName(String nama) {
22         this.nama = nama;
23     }
24
25     public String getName() {
26         return nama;
27     }
28
29     public void setKaki(int jmlKaki) {
30         this.jmlKaki = jmlKaki;
31     }
32
33     public int getKaki() {
34         return jmlKaki;
35     }
36
37     public abstract void displayBinatang();
38 }
```

Class IKarnivora

```
12 public interface IKarnivora {
13     public abstract void displayMakan();
14 }
```

Class IHerbivora

```
12 public interface IHerbivora {
13     public abstract void displayMakan();
14 }
```

Class Singa

```
11 public class Singa extends Binatang implements IKarnivora {
12
13     private String suara;
14     private String warnaBulu;
15
16     public Singa(String nama, int jmlKaki, String suara, String warnaBulu) {
17         super(nama, jmlKaki);
18         this.suara = suara;
19         this.warnaBulu = warnaBulu;
20     }
21
22     @Override
23     public void displayBinatang() {
24         System.out.println("Binatangnya adalah " + super.getNama());
25     }
26
27     @Override
28     public void displayMakan() {
29         System.out.println("Daging dimakan oleh " + super.getNama());
30     }
31
32     public void displayData() {
33         System.out.println("Nama Binatang : " + super.getNama());
34         System.out.println("Memunya Jumlah Kaki : " + super.getKaki());
35         System.out.println("Bunyi Suara : " + this.suara);
36         System.out.println("Warna Bulu : " + this.warnaBulu);
37         System.out.println("=====");
38     }
39 }
```

Class Keledai

```
11 public class Keledai extends Binatang implements IHerbivora {
12
13     private String suara;
14     private String warnaBulu;
15
16     public Keledai(String nama, int jmlKaki, String suara, String warnaBulu) {
17         super(nama, jmlKaki);
18         this.suara = suara;
19         this.warnaBulu = warnaBulu;
20     }
21
22     @Override
23     public void displayBinatang() {
24         System.out.println("Binatangnya adalah " + super.getNama());
25     }
26
27     @Override
28     public void displayMakan() {
29         System.out.println("Rumput dimakan oleh " + super.getNama());
30     }
31
32     public void displayData() {
33         System.out.println("Nama Binatang : " + super.getNama());
34         System.out.println("Memunya Jumlah Kaki : " + super.getKaki());
35         System.out.println("Bunyi Suara : " + this.suara);
36         System.out.println("Warna Bulu : " + this.warnaBulu);
37         System.out.println("=====");
38     }
39 }
```

Class Gorilla

```
11 public class Gorilla extends Binatang implements IHerbivora, IKarnivora{
12
13     private String suara;
14     private String warnaBulu;
15
16     public Gorilla(String nama, int jmlKaki, String suara, String warnaBulu) {
17         super(nama, jmlKaki);
18         this.suara = suara;
19         this.warnaBulu = warnaBulu;
20     }
21
22     @Override
23     public void displayBinatang() {
24         System.out.println("Binatangnya adalah " + super.getNama());
25     }
26
27     @Override
28     public void displayMakan() {
29         System.out.println("Daging dan Rumput dimakan oleh " + super.getNama());
30     }
31
32     public void displayData() {
33         System.out.println("Nama Binatang : " + super.getNama());
34         System.out.println("Memunya Jumlah Kaki : " + super.getKaki());
35         System.out.println("Bunyi Suara : " + this.suara);
36         System.out.println("Warna Bulu : " + this.warnaBulu);
37         System.out.println("=====");
38     }
39 }
```

Class BinatangMain

```
11 public class BinatangMain {
12
13     public static void main(String[] args) {
14
15         Singa s1 = new Singa("Singa", 4, "KUKURUYUKKK ROARRRRR", "Hitam Putih");
16         s1.displayBinatang();
17         s1.displayMakan();
18         s1.displayData();
19
20         Keledai k1 = new Keledai("Keledai", 4, "Kiek-kiek-kiek", "Abu-Abu");
21         k1.displayBinatang();
22         k1.displayMakan();
23         k1.displayData();
24
25         Gorilla g1 = new Gorilla("Gorilla", 2, "UU II U AA JENG JENG PALAPALA JENGJENG", "Hitam");
26         g1.displayBinatang();
27         g1.displayMakan();
28         g1.displayData();
29
30     }
31 }
```


Output

```
Binatangnya adalah Singa
Daging dimakan oleh Singa
Nama Binatang : Singa
Mempunya Jumlah Kaki : 4
Bunyi Suara : KUKURUYUKKKK ROARRRRR
Warna Bulu : Hitam Putih
=====
Binatangnya adalah Keledai
Rumput dimakan oleh Keledai
Nama Binatang : Keledai
Mempunya Jumlah Kaki : 4
Bunyi Suara : Kiek-kiek-kiek
Warna Bulu : Abu-Abu
=====
Binatangnya adalah Gorilla
Daging dan Rumput dimakan oleh Gorilla
Nama Binatang : Gorilla
Mempunya Jumlah Kaki : 2
Bunyi Suara : UU II U AA JENG JENG PALAPALA JENGJENG
Warna Bulu : Hitam
=====
BUILD SUCCESSFUL (total time: 0 seconds)
```