

TUGAS INHARINTANCE PEMROGRAMAN BERBASIS OBJECT

Disusun untuk Memenuhi Tugas Mata Kuliah Pemrograman Berbasis Object yang Dibimbing
oleh YOGA DWITYA PRAMUDITA, S.Kom., M.Cs.



Disusun oleh:

Achmad Farid Alfa Waid

(190411100073)

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS TRUNOJOYO MADURA

2020

1. CODE PROGRAM

Hewan.java

```
public class Hewan {
    public String swimming (String renang) {
        return renang;
    }

    public String flying (String terbang) {
        return terbang;
    }

    public String walking (String jalan) {

        return jalan;
    }

    public String running (String lari) {

        return lari;
    }
}

class Elang extends Hewan {

    public void tingkah_laku(){
        System.out.println("----- HEWAN ELANG -----");
        System.out.println("Elang bernafas : " + this.bernafas("wushhh"));
        System.out.println("Suara Elang terbang : " + super.flying("Wing -
wing"));

        System.out.println("Suara Elang : " + this.bersuara("Kwuaikkk"));
        System.out.println("Elang berkembangbiak dengan cara : " +
this.berkembangbiak("Bertelur"));
        System.out.println("Elang makan : " + this.makan("Daging"));
    }

    public String bernafas(String nafas){
        return nafas;
    }

    public String bersuara(String suara){
        return suara;
    }
}
```

```

    }

    public String berkembangbiak(String KB){
        return KB;
    }

    public String makan(String makanan){
        return makanan;
    }
}

class Bebek extends Hewan{

    public void tingkah_laku(){
        System.out.println("----- HEWAN BEBEK -----");
        System.out.println("Bebek bernafas : " + this.bernafas("huimmm"));
        System.out.println("Suara      Bebek      berenang      :      "      +
super.swimming("Kecipak - kecipuk"));
        System.out.println("Suara      Bebek      :      "      +      this.bersuara("Wekk
wekkk"));

        System.out.println("Bebek      berkembangbiak      dengan      cara      :      "      +
this.berkembangbiak("Bertelur"));
        System.out.println("Bebek makan : " + this.makan("Dedak"));
    }

    public String bernafas(String nafas){
        return nafas;
    }

    public String bersuara(String suara){
        return suara;
    }

    public String berkembangbiak(String KB){
        return KB;
    }

    public String makan(String makanan){
        return makanan;
    }
}

```

```

class Sapi extends Hewan {

    public void tingkah_laku(){
        System.out.println("----- HEWAN SAPI -----");
        System.out.println("Sapi bernafas : " + this.bernafas("ehmmm"));
        System.out.println("Suara Sapi berjalan : " +
super.walking("Klutak - Klutuk"));
        System.out.println("Suara Sapi : " + this.bersuara("Mooooo"));
        System.out.println("Sapi berkembangbiak dengan cara : " +
this.berkembangbiak("Beranak / Melahirkan"));
        System.out.println("Sapi makan : " + this.makan("Rumput"));
    }

    public String bernafas(String nafas){
        return nafas;
    }

    public String bersuara(String suara){
        return suara;
    }

    public String berkembangbiak(String KB){
        return KB;
    }

    public String makan(String makanan){
        return makanan;
    }
}

class Domba extends Hewan {

    public void tingkah_laku(){
        System.out.println("----- HEWAN DOMBA -----");
        System.out.println("Domba bernafas : " +
this.bernafas("werreee"));
        System.out.println("Suara Domba berlari : " +
super.running("gruduk - gruduk"));
        System.out.println("Suara Domba : " + this.bersuara("Mbekkkk"));
        System.out.println("Domba berkembangbiak dengan cara : " +
this.berkembangbiak("Beranak / Melahirkan"));
        System.out.println("Domba makan : " + this.makan("Rumput"));
    }
}

```

```

    }

    public String bernafas(String nafas){
        return nafas;
    }

    public String bersuara(String suara){
        return suara;
    }

    public String berkembangbiak(String KB){
        return KB;
    }

    public String makan(String makanan){
        return makanan;
    }
}

```

Membuat sebuah public class yang bernama Hewan, class tersebut merupakan superclass. Di dalam class tersebut terdapat beberapa method, yaitu method swimming, flying, walking, dan running.

Terdapat 5 subclass yaitu class Elang, Bebek, Sapi, Domba. Masing – masing class tersebut mewarisi class Hewan, dan masing – masing dari class tersebut memiliki method bernafas, bersuara, berkembangbiak, dan makan. Kemudian membuat method tingkah_laku untuk menampilkan beberapa method yang sudah disebutkan di atas, untuk mengakses nilai method dari subclass sendiri menggunakan syntax this, sedangkan untuk superclass menggunakan syntax super

Uji.java

```

public class Uji extends Hewan {

    public static void main(String [] args){

        Elang elang = new Elang();
        elang.tingkah_laku();
        Bebek bebek = new Bebek();
        bebek.tingkah_laku();
        Sapi sapi = new Sapi();
        sapi.tingkah_laku();
    }
}

```

```

        Domba domba = new Domba();
        domba.tingkah_laku();

    }

}

```

Membuat sebuah public class yang bernama Uji, class ini merupakan warisan dari class Hewan. Kemudian membuat object baru dari subclass yang ada di class hewan, dan mengakses method tingkah_laku yang ada di masing – masing object untuk ditampilkan di dalam terminal.

Bergerak.java

```

public interface Bergerak {

    public void maju(int satuan);

    public void mundur(int satuan);

    public void kanan(int satuan);

    public void kiri(int satuan);

}

class Kerbau implements Bergerak {

    public void maju(int satuan){
        System.out.println("----- HEWAN KERBAU -----");
        System.out.println("Kerbau bergerak maju " + satuan + " langkah");
    }

    public void mundur(int satuan){
        System.out.println("Kerbau bergerak mundur " + satuan + " langkah");
    }

    public void kanan(int satuan){
        System.out.println("Kerbau bergerak ke kanan " + satuan + " langkah");
    }

    public void kiri(int satuan){
        System.out.println("Kerbau bergerak ke kiri " + satuan + " langkah");
    }

}

```

```

        public void makan(String makanan){
            System.out.println("Kerbau memakan : " + makanan);
        }
    }

class Mobil implements Bergerak {

    public void maju(int satuan){
        System.out.println("----- MOBIL -----");
        System.out.println("Mobil bergerak maju " + satuan + " km/jam");
    }

    public void mundur(int satuan){
        System.out.println("Mobil bergerak mundur " + satuan + " km/jam");
    }

    public void kanan(int satuan){
        System.out.println("Mobil bergerak ke kanan " + satuan + " km/jam");
    }

    public void kiri(int satuan){
        System.out.println("Mobil bergerak ke kiri " + satuan + " km/jam");
    }

    public void isiBensin (int satuan){
        System.out.println("Mobil mengisi bensin sebanyak " + satuan + " liter");
    }

}

class Robot implements Bergerak {

    public void maju(int satuan){
        System.out.println("----- ROBOT -----");
        System.out.println("Robot bergerak maju " + satuan + " m/detik");
    }

    public void mundur(int satuan){
        System.out.println("Robot bergerak mundur " + satuan + " m/detik");
    }

}

```

```

    public void kanan(int satuan){
        System.out.println("Robot bergerak ke kanan " + satuan + " m/detik");
    }

    public void kiri(int satuan){
        System.out.println("Robot bergerak ke kiri " + satuan + " m/detik");
    }

    public void baterai(String daya){
        System.out.println("Robot memakai baterai bersumber daya " + daya);
    }
}

```

Membuat sebuah public class interface yang bernama Bergerak, class tersebut merupakan superclass. Di dalam class tersebut terdapat beberapa method, yaitu method maju, mundur, kanan, dan kiri.

Selanjutnya terdapat tiga subclass yaitu class Kerbau, Mobil, dan Robot. Masing – masing class tersebut mewarisi superclass Bergerak. Sehingga dapat mengakses method maju, mundur, kanan, dan kiri. Kemudian pada subclass Kerbau terdapat method makan, untuk subclass Mobil terdapat method isiBensin, dan pada subclass Robot terdapat method baterai

Uji2.java

```

public class Uji2 {

    public static void main(String[] args) {

        Kerbau kerbau = new Kerbau();

        kerbau.maju(5);

        kerbau.mundur(10);

        kerbau.kanan(15);

        kerbau.kiri(20);

        kerbau.makan("Rumput");

        Mobil mobil = new Mobil();
    }
}

```



```
        mobil.maju(3);

        mobil.mundur(6);

        mobil.kanan(9);

        mobil.kiri(12);

        mobil.isiBensin(5);

        Robot robot = new Robot();

        robot.maju(2);

        robot.mundur(4);

        robot.kanan(6);

        robot.kiri(8);

        robot.baterai("listrik");

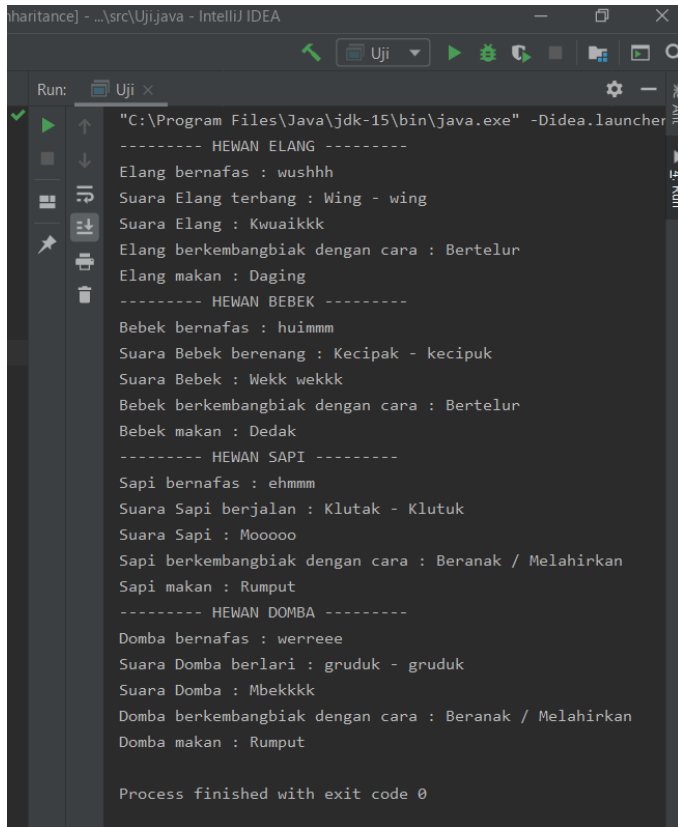
    }

}
```

Membuat sebuah public class yang bernama Uji, pada class ini akan membuat object baru dari masing – masing subclass yang ada di class Bergerak, kemudian mengakses method – method yang ada di masing – masing class tersebut untuk ditampilkan di dalam terminal.

2. HASIL RUNNING PROGRAM

1) Uji.java

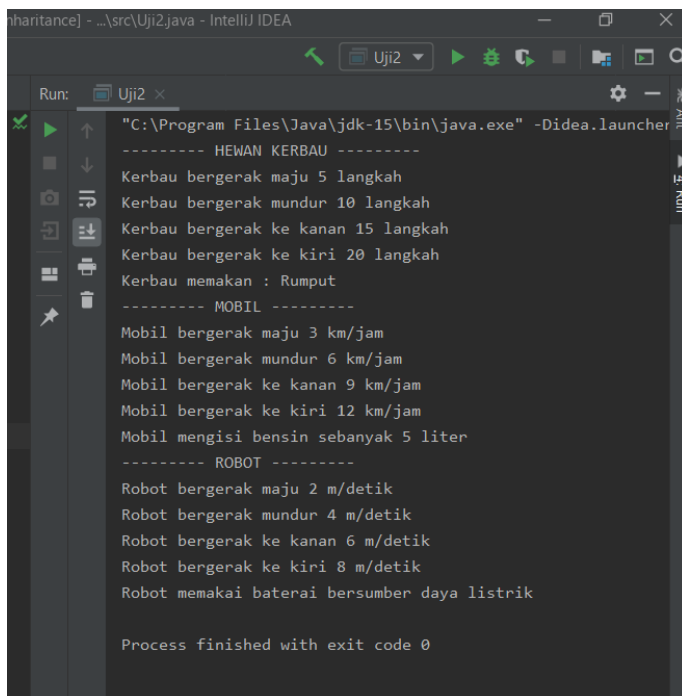


The screenshot shows the IntelliJ IDEA Run console for a program named 'Uji'. The console output is as follows:

```
"C:\Program Files\Java\jdk-15\bin\java.exe" -Didea.launcher
----- HEWAN ELANG -----
Elang bernafas : wushhh
Suara Elang terbang : Wing - wing
Suara Elang : Kwuaikkk
Elang berkembangbiak dengan cara : Bertelur
Elang makan : Daging
----- HEWAN BEBEK -----
Bebek bernafas : huimmm
Suara Bebek berenang : Kecipak - kecipuk
Suara Bebek : Wekk wekkk
Bebek berkembangbiak dengan cara : Bertelur
Bebek makan : Dedak
----- HEWAN SAPI -----
Sapi bernafas : ehmmm
Suara Sapi berjalan : Klutak - Klutak
Suara Sapi : Mooooo
Sapi berkembangbiak dengan cara : Beranak / Melahirkan
Sapi makan : Rumput
----- HEWAN DOMBA -----
Domba bernafas : werneee
Suara Domba berlari : gruduk - gruduk
Suara Domba : Mbekkkk
Domba berkembangbiak dengan cara : Beranak / Melahirkan
Domba makan : Rumput

Process finished with exit code 0
```

2) Uji2.java



The screenshot shows the IntelliJ IDEA Run console for a program named 'Uji2'. The console output is as follows:

```
"C:\Program Files\Java\jdk-15\bin\java.exe" -Didea.launcher
----- HEWAN KERBAU -----
Kerbau bergerak maju 5 langkah
Kerbau bergerak mundur 10 langkah
Kerbau bergerak ke kanan 15 langkah
Kerbau bergerak ke kiri 20 langkah
Kerbau memakan : Rumput
----- MOBIL -----
Mobil bergerak maju 3 km/jam
Mobil bergerak mundur 6 km/jam
Mobil bergerak ke kanan 9 km/jam
Mobil bergerak ke kiri 12 km/jam
Mobil mengisi bensin sebanyak 5 liter
----- ROBOT -----
Robot bergerak maju 2 m/detik
Robot bergerak mundur 4 m/detik
Robot bergerak ke kanan 6 m/detik
Robot bergerak ke kiri 8 m/detik
Robot memakai baterai bersumber daya listrik

Process finished with exit code 0
```