

LAPORAN TUGAS
PEMROGRAMAN BERBASIS OBJEK
PERTEMUAN PERTAMA 22 AGUSTUS 2023



UIN SUNAN AMPEL
S U R A B A Y A

DOSEN PEMBIMBING

Bayu Adhi Nugroho, Ph.D.

(197905182014031001)

DISUSUN OLEH

Mochamad Roiyan Rintiarno

(09020622033)

UIN SUNAN AMPEL SURABAYA

TAHUN 2023

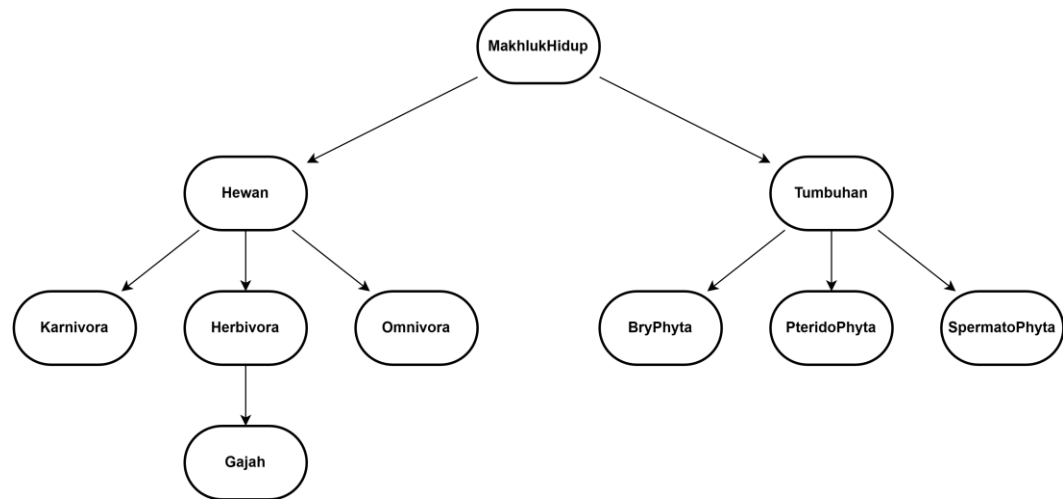
1. Tugas

→ Membuat kelas MakhlukHidup dan turunannya, dengan contoh satu makhluk hidup

2. Pembahasan dan Isi

a. Persiapan dan Langkah-langkah

1. Membuat Diagram



Dari diagram tersebut dapat diketahui jika parent class yang akan dibuat adalah MakhlukHidup, lalu kemudian kedua class tersebut dikelompokkan atau dibagi kembali dengan menambahkan Hewan dan Tumbuhan. Kemudian pada hewan akan dikelompokkan sesuai dengan makanannya, yakni karnivora (pemakan daging) herbivora (pemakan tumbuhan) dan omnivora (pemakan segala). Lalu pada tumbuhan juga dibagi kembali menjadi tiga, yakni BryPhyta (tumbuhan lumut), PteridoPhyta (tumbuhan paku), dan SpermatoPhyta (tumbuhan berbiji). Dan yang terakhir, karena output yang diinginkan adalah gajah, maka kelas Gajah akan diletakkan dibawah kelas Herbivora. Lalu selanjutnya kelas tersebut akan dipanggil melalui kelas lain diluar package MakhlukHidup tersebut agar pembaca sekalian dapat mengetahui perbedaan public, private, dan protected.

2. Membuat Kelas Parent dengan nama MakhlukHidup

```
1 package MakhlukHidup;
2
3 import java.util.ArrayList;
4
5 public class MakhlukHidup {
6     private String nama;
7     private String asal;
8     private String namaFamily;
9     private String bahasaLatin;
10    private String subSpecies;
11    protected String jenisMakanan;
12    protected ArrayList<String> ciriCiri = new ArrayList<String>();
13
14    // nama
15    public String getNama() {
16        return nama;
17    }
18
19    public void setNama(String nama) {
20        this.nama = nama;
21    }
22
23    // asal
24    public String getAsal() {
25        return asal;
26    }
27
28    public void setAsal(String asal) {
29        this.asal = asal;
30    }
31
32    // namaFamily
33    public String getNamaFamily() {
34        return namaFamily;
35    }
36
37    public void setNamaFamily(String namaFamily) {
38        this.namaFamily = namaFamily;
39    }
40
41    // bahasaLatin
42    public String getBahasaLatin() {
43        return bahasaLatin;
44    }
45
46    public void setBahasaLatin(String bahasaLatin) {
47        this.bahasaLatin = bahasaLatin;
48    }
49
50    // subSpecies
51    public String getSubSpecies() {
52        return subSpecies;
53    }
54
55    public void setSubSpecies(String subSpecies) {
56        this.subSpecies = subSpecies;
57    }
58 }
59
```

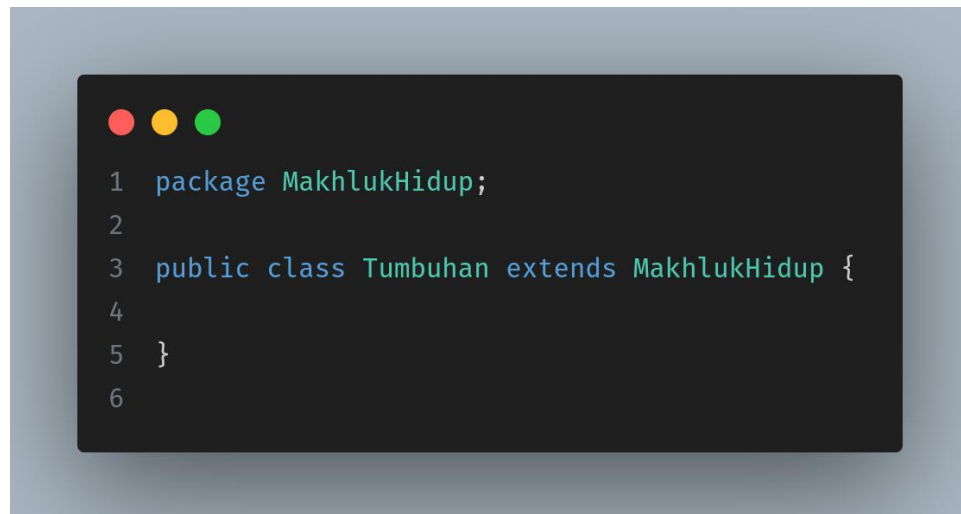
Kelas MakhlukHidup adalah parent dari kelas-kelas setelahnya, terdapat beberapa atribut, yakni nama, asal, namaFamily, bahasaLatin, dan subSpecies yang bersifat private (hanya dapat diakses didalam kelas saja), sehingga diperlukan method untuk mengaksesnya secara publik, hal ini dinamakan dengan getter dan setter. Dimana atribut nama memiliki method getter yakni getNama dan setter yakni setNama, lalu kemudian atribut asal yang memiliki method getter getAsal dan setter yakni setAsal, kemudian atribut namaFamily yang memiliki method getter getNamaFamily dan setter yakni setNamaFamily, selanjutnya ada atribut bahasaLatin yang memiliki method getter getBahasaLatin dan setter setBahasaLatin, dan yang terakhir ada atribut subSpecies yang memiliki method getter getSubSpecies dan setter setSubSpecies. Lalu kemudian ada atribut jenisMakanan dan ciriCiri yang bersifat protected, yakni atribut tersebut hanya dapat diakses didalam kelas itu sendiri dan didalam kelas turunannya. Pada kelas ini tidak dibuatkan setter dan getternya.

3. Membuat Kelas Child pertama, yakni kelas Hewan dan kelas Tumbuhan
 - a. Kelas Hewan

```
1 package MakhlukHidup;
2
3 import java.util.ArrayList;
4
5 public class Hewan extends MakhlukHidup {
6     public int jumlahKaki = 0;
7
8     public ArrayList<String> getCiriCiri() {
9         return ciriCiri;
10    }
11
12    public void addCiriCiri(String ciriCiri) {
13        this.ciriCiri.add(ciriCiri);
14    }
15 }
16
```

Pada gambar diatas, kelas Hewan mewarisi kelas MakhlukHidup, hal ini ditandai dengan adanya kata kunci extends yang berarti setiap metod atau atribut yang berada pada kelas Makhluk hidup yang bersifat public/protected akan dapat digunakan pada kelas Hewan, tanpa menduplikat kodenya. Kelas Hewan memiliki 1 atribut yakni jumlahKaki yang bersifat publik, sehingga atribut tersebut dapat diakses oleh seluruh yang memanggilnya. Lalu terdapat setter dan getter atribut ciriCiri yakni getCiriCiri dan addCiriCiri. Pada kelas Hewan, kita tidak perlu menambahkan atribut ciriCiri karena hal tersebut telah ditambahkan dalam kelas induknya (parent) dan bersifat protected sehingga dapat dipanggil di kelas turunannya.

b. Kelas Tumbuhan

A screenshot of a code editor with a dark background and light blue text. The code is as follows:

```
1 package MakhlukHidup;  
2  
3 public class Tumbuhan extends MakhlukHidup {  
4  
5 }  
6
```

Pada gambar tersebut, kelas Tumbuhan akan mewarisi kelas MakhlukHidup sehingga semua method dan atribut yang berada pada kelas MakhlukHidup juga akan terdapat didalam kelas Tumbuhan.

4. Membuat Kelas Child untuk Kelas Hewan, yakni kelas Karnivora, Herbivora, dan Omnivora
 - a. Kelas Karnivora

```
1 package MakhlukHidup;
2
3 public class Karnivora extends Hewan {
4
5 }
6
```

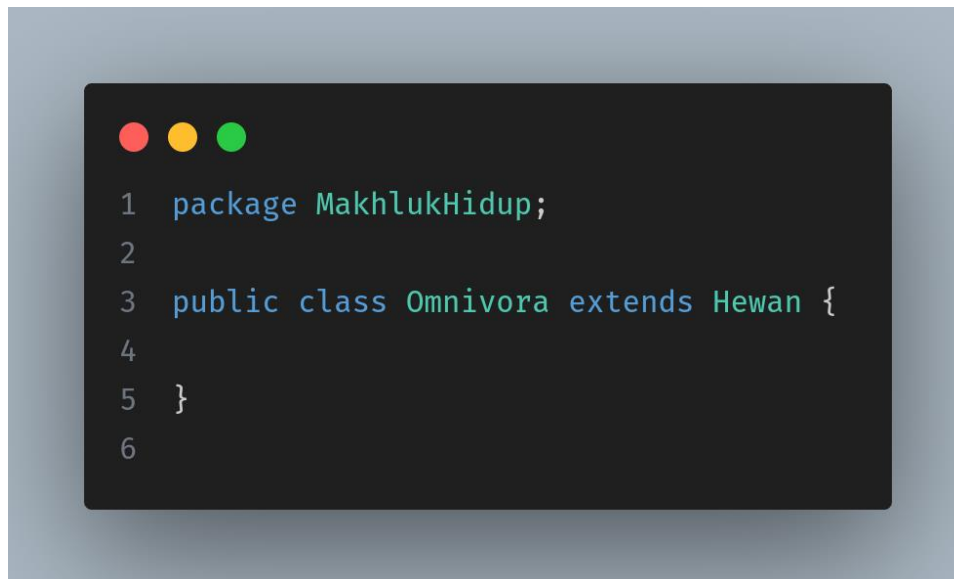
Pada gambar tersebut, kelas Karnivora akan mewarisi kelas Hewan sehingga semua method dan atribut yang berada pada kelas Hewan juga akan terdapat didalam kelas Karnivora.

b. Kelas Herbivora

```
1 package MakhlukHidup;
2
3 public class Herbivora extends Hewan {
4
5     public Herbivora() {
6         this.jenisMakanan = "Herbivora";
7     }
8
9     public String getJenisMakanan() {
10         return jenisMakanan;
11     }
12 }
13
```

Pada gambar tersebut, kelas Herbivora akan mewarisi kelas Hewan sehingga semua method dan atribut yang berada pada kelas Hewan juga akan terdapat didalam kelas Herbivora. Di dalam kelas Herbivora juga terdapat method constructor, yakni method yang akan langsung dipanggil saat kita memanggil kelas tersebut, sebagaimana contoh pada kelas Herbivora diatas, yakni ketika kelas tersebut dipanggil, maka akan langsung meng-set atribut jenisMakanan yang terdapat pada kelas parent (dalam hal ini method tersebut terdapat pada kelas MakhlukHidup), sehingga atribut jenisMakanan akan langsung memiliki value “Herbivora” yang bertipe data String. Lalu didalam kelas tersebut juga terdapat method getter yang bernama getJenisMakanan.

c. Kelas Omnivora

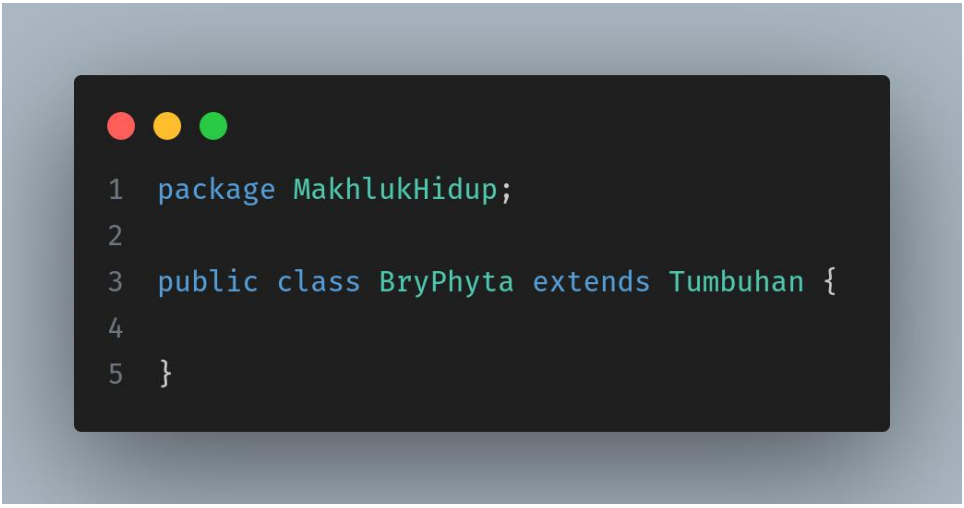
A screenshot of a code editor with a dark background and light-colored text. The code is written in Java and defines a class named Omnivora that extends the Hewan class. The code is as follows:

```
1 package MakhlukHidup;
2
3 public class Omnivora extends Hewan {
4
5 }
6
```

Pada gambar tersebut, kelas Omnivora akan mewarisi kelas Hewan sehingga semua method dan atribut yang berada pada kelas Hewan juga akan terdapat didalam kelas Omnivora.

5. Membuat Kelas Child untuk kelas Tumbuhan, yakni kelas BryPhyta, PteridoPhyta, SpermatoPhyta

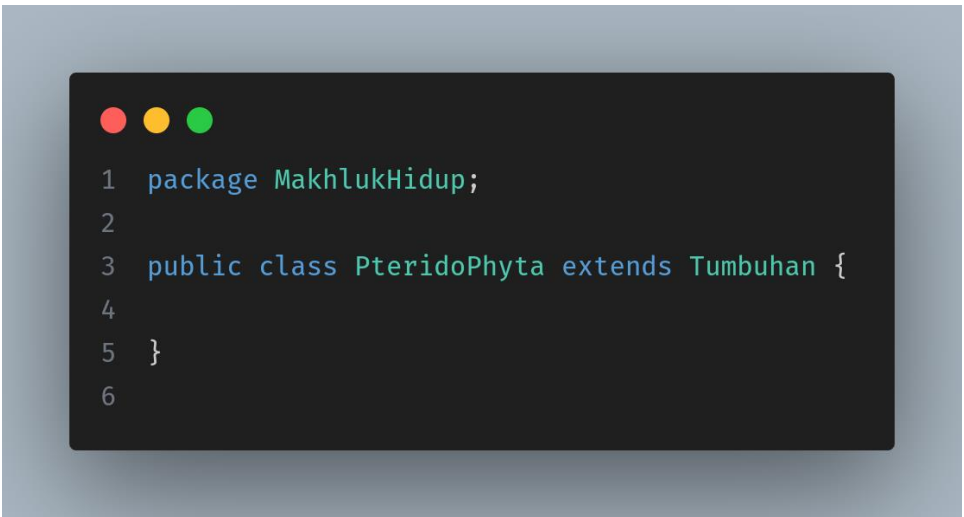
a. Kelas BryPhyta

A screenshot of a code editor with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. The code is written in a light blue font and consists of five lines. Line 1: `package MakhlukHidup;` Line 2: (empty line) Line 3: `public class BryPhyta extends Tumbuhan {` Line 4: (empty line) Line 5: `}`

```
1 package MakhlukHidup;  
2  
3 public class BryPhyta extends Tumbuhan {  
4  
5 }
```

Pada gambar tersebut, kelas BryPhyta akan mewarisi kelas Tumbuhan sehingga semua method dan atribut yang berada pada kelas Tumbuhan juga akan terdapat didalam kelas BryPhyta.

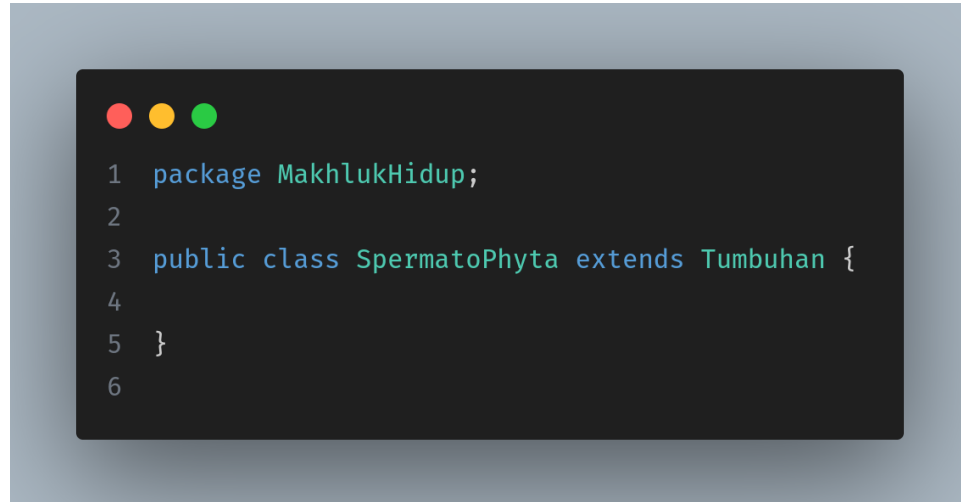
b. Kelas PteridoPhyta

A screenshot of a code editor with a dark background and three colored window control buttons (red, yellow, green) in the top left corner. The code is written in a light blue font and consists of six lines. Line 1: `package MakhlukHidup;` Line 2: (empty line) Line 3: `public class PteridoPhyta extends Tumbuhan {` Line 4: (empty line) Line 5: `}` Line 6: (empty line)

```
1 package MakhlukHidup;  
2  
3 public class PteridoPhyta extends Tumbuhan {  
4  
5 }  
6
```

Pada gambar tersebut, kelas PteridoPhyta akan mewarisi kelas Tumbuhan sehingga semua method dan atribut yang berada pada kelas Tumbuhan juga akan terdapat didalam kelas PteridoPhyta.

c. Kelas SpermatoPhyta



Pada gambar tersebut, kelas SpermatoPhyta akan mewarisi kelas Tumbuhan sehingga semua method dan atribut yang berada pada kelas Tumbuhan juga akan terdapat didalam kelas SpermatoPhyta.

6. Membuat package baru bernama Output dengan nama kelas CobaOutput untuk mencoba package MakhlukHidup sebelumnya, menggunakan package baru bertujuan agar dapat melihat hasil implementasi dari parameter protected

```

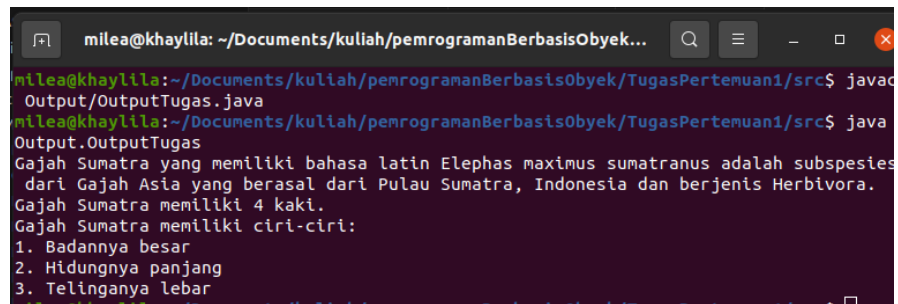
1 package Output;
2
3 import java.util.ArrayList;
4 import MakhlukHidup.Herbivora;
5
6 public class OutputTugas {
7     public static void main(String[] args) {
8         Herbivora gajah = new Herbivora();
9         gajah.setNama("Gajah Sumatra");
10        gajah.setAsal("Pulau Sumatra, Indonesia");
11        gajah.setBahasaLatin("Elephas maximus sumatranus");
12        gajah.setSubSpecies("Gajah Asia");
13        gajah.jumlahKaki = 4;
14        gajah.addCiriCiri("Badannya besar");
15        gajah.addCiriCiri("Hidungnya panjang");
16        gajah.addCiriCiri("Telinganya lebar");
17
18        // Sysout
19        System.out.println(gajah.getNama()
20            + " yang memiliki bahasa latin " + gajah.getBahasaLatin()
21            + " adalah subspesies dari " + gajah.getSubSpecies()
22            + " yang berasal dari " + gajah.getAsal()
23            + " dan berjenis " + gajah.getJenisMakanan() + ".");
24        System.out.println(gajah.getNama()
25            + " memiliki " + gajah.jumlahKaki + " kaki.");
26        System.out.println(gajah.getNama()
27            + " memiliki ciri-ciri: ");
28        ArrayList<String> ciriCiri = gajah.getCiriCiri();
29        for (int i = 0; i < ciriCiri.size(); i++) {
30            System.out.println((i + 1) + ". " + ciriCiri.get(i));
31        }
32    }
33 }
34

```

Pada kelas OutputTugas, langkah awal untuk menggunakan kelas Herbivora adalah dengan memanggil kelasnya terlebih dahulu. Pada java, cara memanggil kelas yakni 'NamaKelas namaVariabel = new Namakelas();'. Setelah itu, untuk menambahkan nama tinggal memanggil method untuk meng-set atribut nama, yakni setName. Lalu untuk menambahkan asal tinggal memanggil method untuk meng-set atribut asal, yakni setAsal. Kemudian hal yang sama juga dijalankan pada atribut bahasaLatin dan subSpecies, yakni tinggal memanggil method setter-nya saja (setBahasaLatin dan setSubSpecies). Lalu kemudian untuk menambahkan ciri-ciri, yakni tinggal memanggil setternya, yaitu addCiriCiri. Dan pada atribut jumlahKaki, karena atribut tersebut bersifat publik, sehingga kita boleh tidak perlu menggunakan getter dan setter, melainkan hanya dengan menggunakan 'gajah.jumlahKaki = 4' (namaVariabel.atribut = value). Lalu pada

saat pemanggilan pun sama, hanya tinggal memanggil method getternya saja, seperti atribut nama yang method getternya adalah getName dan atribut asal yang method getternya adalah getAsal.

7. Hasil/Output



```
milea@khaylila: ~/Documents/kuliah/pemrogramanBerbasisObyek...
milea@khaylila:~/Documents/kuliah/penrogramanBerbasisObyek/TugasPertemuan1/src$ javac
Output/OutputTugas.java
milea@khaylila:~/Documents/kuliah/penrogramanBerbasisObyek/TugasPertemuan1/src$ java
Output/OutputTugas
Gajah Sumatra yang memiliki bahasa latin Elephas maximus sumatranus adalah subspesies
dari Gajah Asia yang berasal dari Pulau Sumatra, Indonesia dan berjenis Herbivora.
Gajah Sumatra memiliki 4 kaki.
Gajah Sumatra memiliki ciri-ciri:
1. Badannya besar
2. Hidungnya panjang
3. Telinganya lebar
```