

LAPORAN PRAKTIKUM

PEMROGRAMAM BERORIENTASI OBJEK

Modul : 6 Tanggal praktikum : 17-05-2023 (*Format: DD-MM-YYYY*)

Nama : Muhamad Fahmi

NIM : 20220810029

Kelas : TINFC-2022-02 (*contoh: TINFC-2022-01*)

JUDUL/TEMA MODUL

1. Tujuan Pembelajaran

Tuliskan tujuan pembelajaran/praktikum. Sesuaikan dengan tujuan pembelajaran pada modul praktikum.

1. Memahami Polimorphism
2. Memahami Konsep Polimorphism
3. Mampu Membuat Program Java Polimorphism

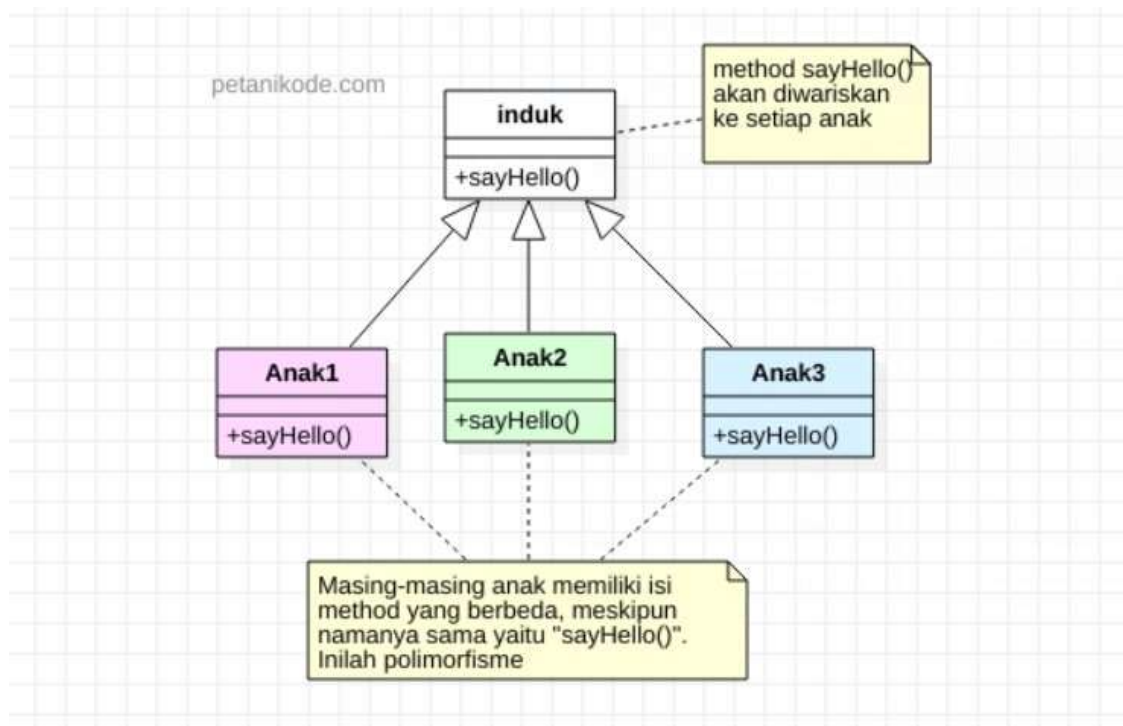
2. Dasar Teori

Pada bagian ini, silahkan kalian tulis/tambahkan materi pendukung untuk kegiatan praktikumnya. Tambahkan gambar / tabel/ flowchart apabila diperlukan.

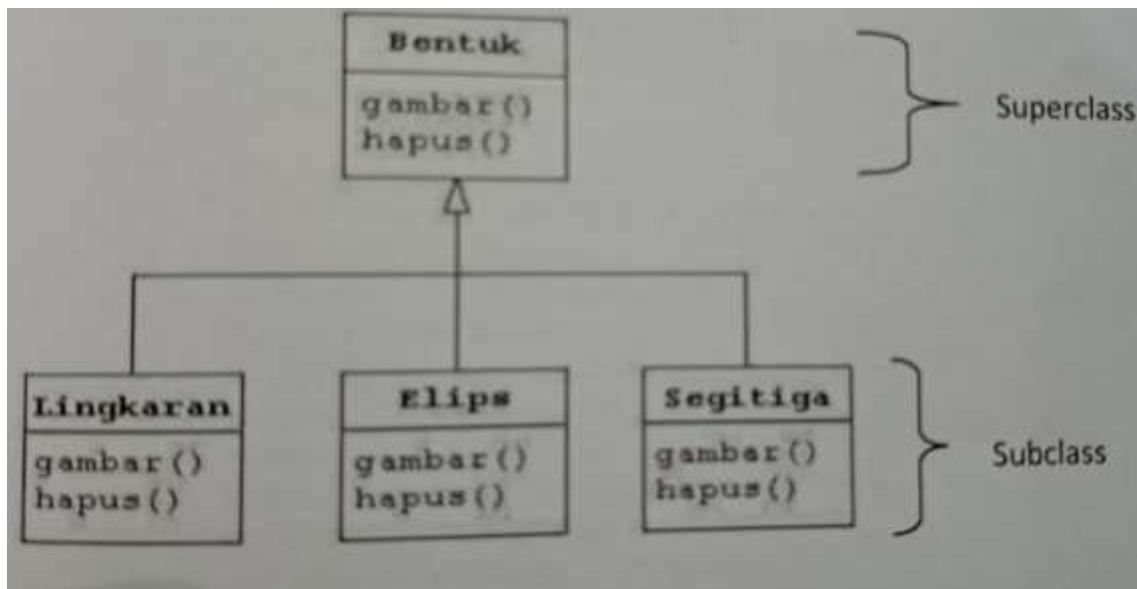
Polymorphism

Polymorphism mempunyai makna sesuatu yang memiliki banyak bentuk, yaitu memiliki nama sama, tetapi memiliki kelakuan (behaviour) yang berbeda.

Poly artinya banyak, morfisme artinya bentuk. Polimorfisme (bahasa inggris polymorphism) adalah sebuah prinsip dalam biologi di mana organisme atau spesies dapat memiliki banyak bentuk atau tahapan (Muhardian, 2019).



Contoh Gambar Polimorphism diambil dari (Muhardian, 2019).



Perhatikan gambar diagram diatas!

Class Bentuk merupakan class induk (*superclass*) dari class Lingkaran, Elips, dan Segitiga mempunyai method **gambar()** dan **hapus()**. Class-class anak (*subclass*) juga mempunyai method **gambar()** dan **hapus()**. Meskipun keempat class tersebut mempunyai nama method yang sama, tetapi isi (source code / code yang dilakukan / output) dari masing-masing method tersebut berbeda.

Jika kita menginginkan sebuah objek yang dapat memanggil setiap method (yaitu method gambar & hapus) yang ada pada setiap class (pada superclass maupun subclass), maka gunakanlah teknik **Polymorphism**. Polymorphism hanya berlaku pada method dan tidak berlaku pada atribut.

Untuk mendapatkan operasi polymorphism dari suatu method, maka method tersebut haruslah merupakan method yang ada di class induk (lihat diagram diatas bahwa method gambar() & hapus()), selain terdapat class-class turunan class Bentuk, juga terdapat di class Bentuk).

3. Hasil dan Pembahasan

- Bagian ini berisi hasil praktikum yang telah anda kerjakan yang meliputi *pretest*, tahapan praktikum, tugas, dan *posttest*.
- Lampirkan hasil pekerjaan berupa tangkapan layar (*schreenshoot*) kegiatan praktikum sesuai dengan langkah kerja yang ada.

PRETEST

1. Gambarkan contoh diagram inheritance dalam kehidupan sehari-hari!

E. PRAKTIKUM

PRAKTIKUM 1

1. Buatlah Project Baru di Netbeans dengan nama modul5PBO
2. Buat File Empty Java File Baru
3. Ketikkan Source Code berikut ini dan simpan dengan nama file **TesBentuk.java**

```
package modul5PBO;

class Bentuk{
    public void gambar(){System.out.println("Menggambar");}
    public void hapus(){System.out.println("Menghapus gambar\n");}
}

class Lingkaran extends Bentuk{
    public void gambar(){System.out.println("Gambar Lingkaran\n");}
    public void hapus(){System.out.println("Hapus lingkaran\n");}
}

class Elips extends Bentuk{
    public void gambar(){System.out.println("Gambar Elips\n");}
    public void hapus(){System.out.println("Hapus Elips\n");}
}

class Segitiga extends Bentuk{
    public void gambar(){System.out.println("Gambar Segitiga\n");}
    public void hapus(){System.out.println("Hapus lingkaran\n");}
}

public class TesBentuk{
    public static void main(String[] args) {
        Bentuk b = new Bentuk();
        Lingkaran l = new Lingkaran();
        Elips e = new Elips();
        Segitiga s = new Segitiga();

        System.out.println("Bentuk");
        b.gambar();b.hapus();

        System.out.println("Bentuk Lingkaran");
        l.gambar();l.hapus();

        System.out.println("Bentuk Elips");
        e.gambar();e.hapus();

        System.out.println("Bentuk Segitiga");
        s.gambar();s.hapus();
    }
}
```

4. Jalankan file **TesBentuk.java** tersebut, amati hasilnya

```
Output - PraktikumPBO (run)

run:
Bentuk
Menggambar
Menghapus gambar

Bentuk Lingkaran
Gambar lingkaran

Hapus lingkaran

Bentuk Elips
Gambar Elips

Hapus Elips

Bentuk Segitiga
Gambar Segitiga

Hapus lingkaran

BUILD SUCCESSFUL (total time: 0 seconds)
```

Pembahasan:

Pada praktikum 1 ini hasil dari program ini penerapan Polimorphism dimana Bentuk sebagai superclass dan lingkaran, elips, dan segitiga sebagai subclass pada subclass ditandai dengan ***extends***. Lalu dicetak pada class TesBentuk

PRAKTIKUM 2

1. Buat File Empty Java File Baru dengan nama **BentukWajah.java**
2. Ketikkan Soucre Code berikut ini:

```

package modul6PBO;

import java.*;
import java.io.*;

class BentukWajah{
    public String respons(){
        return("Perhatikan reaksi wajah saya");
    }
}

class senyum extends BentukWajah{
    public String respons(){
        return("Senyum karna senang");
    }
}

class Tertawa extends BentukWajah{
    public String respon(){
        return("Tertawa karna Gembira");
    }
}

class Marah extends BentukWajah{
    public String respon(){
        return("Kemarahan disebabkan bertengkar");
    }
}

class Sedih extends BentukWajah{
    public String respon(){
        return("Kemarahan disebabkan cemburu");
    }
}
}

```

3. Buat File Empty Java File Baru dengan nama **Programpolimorphism.java**
4. Ketikkan Source Code Berikut ini:

```

package modul6PBO;

public class Programpolimorphism{
    public static void main(String[] args) {
        System.out.println("\n Program Polimorphism");
        System.out.println("=====");

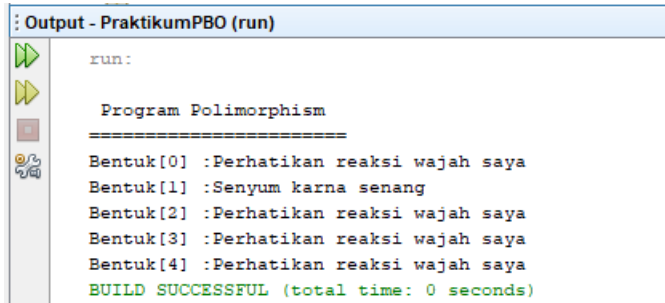
        //      Membuat Objek-Objek raut (ekspresi) dari wajah
        BentukWajah objekbentuk = new BentukWajah();
        senyum objeksenyum = new senyum();
        Tertawa objektertawa = new Tertawa();
        Marah objekmarah = new Marah();
        Sedih objeksedih = new Sedih();

        BentukWajah [] Bentuk = new BentukWajah[5];
        Bentuk[0]= objekbentuk;
        Bentuk[1]= objeksenyum;
        Bentuk[2]= objektertawa;
        Bentuk[3]= objekmarah;
        Bentuk[4]= objeksedih;

        System.out.println("Bentuk[0] :" + Bentuk[0].respons());
        System.out.println("Bentuk[1] :" + Bentuk[1].respons());
        System.out.println("Bentuk[2] :" + Bentuk[2].respons());
        System.out.println("Bentuk[3] :" + Bentuk[3].respons());
        System.out.println("Bentuk[4] :" + Bentuk[4].respons());
    }
}

```

5. Jalankan file **ProgramPolimorphism.java** tersebut, amati hasilnya



```
run:
Program Polimorphism
=====
Bentuk[0] :Perhatikan reaksi wajah saya
Bentuk[1] :Senyum karna senang
Bentuk[2] :Perhatikan reaksi wajah saya
Bentuk[3] :Perhatikan reaksi wajah saya
Bentuk[4] :Perhatikan reaksi wajah saya
BUILD SUCCESSFUL (total time: 0 seconds)
```

Pembahasan:

Pada praktikum 2 ini hasil dari program ini penerapan Polimorphism dimana BentukWajah sebagai superclass dan senyum, tertawa, marah, dan sedih sebagai subclass pada subclass ditandai dengan **extends**. Lalu dicetak pada file ProgramPolimorphism.java

PRAKTIKUM 3

1. Buat File Empty Java File Baru dengan nama **Hewan.java**
2. Ketikkan Soucre Code berikut ini:

```

package modul6PBO;

public class Hewan{

    String jenis,cari;
    public void suara(){System.out.println("Suara Hewan");}
    public void Berjalan(){System.out.println("Cara berjalan hewan");}
}

class Singa extends Hewan{

    String jenis,cari;
    public void suara(){System.out.println("Wou wou");}
    public void Berjalan(){
        System.out.println("Berlarin");
        System.out.println("=====");
    }
}

class Elang extends Hewan{

    String jenis,cari;
    public void suara(){System.out.println("leokkkkkkk");}
    public void Berjalan(){
        System.out.println("Terbang");
        System.out.println("=====");
    }
}

class Lebah extends Hewan{

    String jenis,cari;
    public void suara(){System.out.println("Wungggg");}
    public void Berjalan(){
        System.out.println("Terbang");
        System.out.println("=====");
    }
}

class Paus extends Hewan{

    String jenis,cari;
    public void suara(){System.out.println("Boom");}
    public void Berjalan(){
        System.out.println("Berenang");
        System.out.println("=====");
    }
}

```

3. Buat File Empty Java File Baru dengan nama **DemoHewan.java**
4. Ketikkan Source Code Berikut ini:

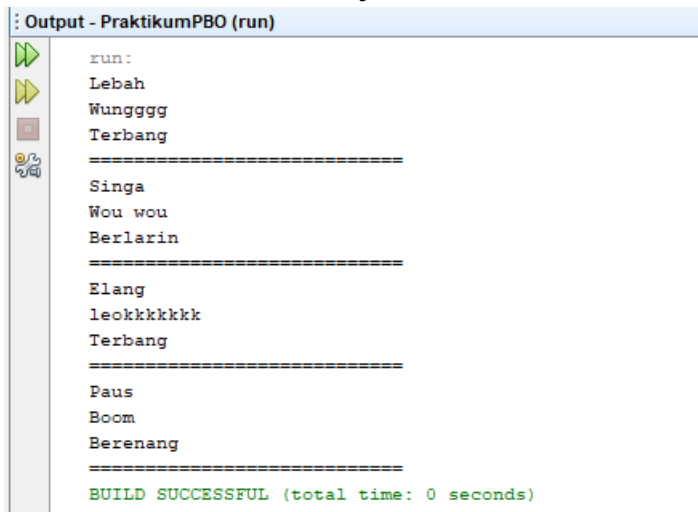

```

package modul6PBO;

public class DemoHewan{
    public static void main(String[] args) {
        Lebah HewanLebah = new Lebah();
        Paus HewanPaus = new Paus();
        Singa HewanSinga = new Singa();
        Elang HewanElang = new Elang();
        Hewan h;
        System.out.println("Lebah");
        h = HewanLebah; h.suara(); h.Berjalan();
        System.out.println("Singa");
        h = HewanSinga; h.suara(); h.Berjalan();
        System.out.println("Elang");
        h = HewanElang; h.suara(); h.Berjalan();
        System.out.println("Paus");
        h = HewanPaus; h.suara(); h.Berjalan();
    }
}

```

5. Jalankan file **DemoHewan.java** tersebut, amati hasilnya



```

: Output - PraktikumPBO (run)

run:
Lebah
Wungggg
Terbang
=====
Singa
Wou wou
Berlarin
=====
Elang
leokkkkkkk
Terbang
=====
Paus
Boom
Berenang
=====
BUILD SUCCESSFUL (total time: 0 seconds)

```

Pembahasan:

Pada praktikum 3 ini hasil dari program ini penerapan Polimorphism dimana Hewan sebagai superclass dan Singa, Elang, Lebah, dan Paus sebagai subclass pada subclass ditandai dengan **extends**. Lalu dicetak pada file DemoHewan.java

POSTTEST

1. Buatlah Program **Polimorphism** dengan case studi yang berbeda dari praktikum diatas.

```
import java.*; import
java.io.*;
/*class induk dengan nama PabrikanMotor*/class PabrikanMotor{
    //Membuat sebuah method respons pada class induk dimanamengembalikan
    public String respons()
    {return("Top speed pabrikan motor\n");
    }
}
/*Membuat class-class lain dengan turunan dari class pabrikanmotor*/
class Yamaha extends PabrikanMotor{public String
    respons()
    {return("299 km/jam \n");}
}
class Honda extends PabrikanMotor{public String
    respons()
    {return("301 km/jam \n");}
}
class Suzuki extends PabrikanMotor{public String
    respons()
    {return("299 km/jam \n");}
}
class Kawasaki extends PabrikanMotor{public String
    respons()
    {return("329 km/jam \n");}
}
```

Programpolimorphism2.java

```
public class Programpolimorphism2{
    public static void main(String []args)
    {
        System.out.println("\n Program polimorphs");
        System.out.println("====="); PabrikanMotor
        objpabrikan=new PabrikanMotor();Yamaha objyamaha=new
        Yamaha();
        Honda objhonda=new Honda(); Suzuki
        objsuzuki=new Suzuki();
        Kawasaki objkawasaki=new Kawasaki(); PabrikanMotor
        []Pabrikan=new PabrikanMotor[5];Pabrikan[0]=objpabrikan;
        Pabrikan[1]=objyamaha;
        Pabrikan[2]=objhonda;
        Pabrikan[3]=objsuzuki;
        Pabrikan[4]=objkawasaki;

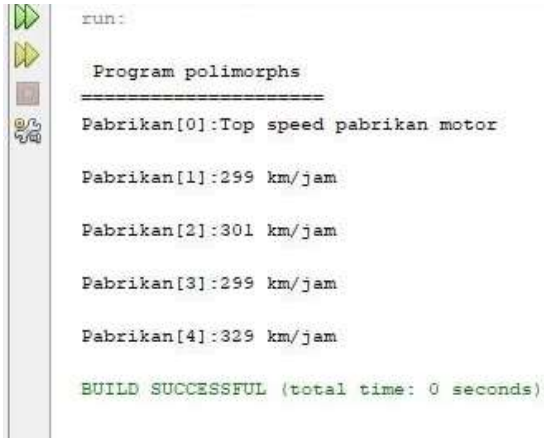
        System.out.println("Pabrikan[0]:"+Pabrikan[0].respons());

        System.out.println("Pabrikan[1]:"+Pabrikan[1].respons());

        System.out.println("Pabrikan[2]:"+Pabrikan[2].respons());

        System.out.println("Pabrikan[3]:"+Pabrikan[3].respons());

        System.out.println("Pabrikan[4]:"+Pabrikan[4].respons());
    }
}
```



```
run:
Program polimorphs
=====
Pabrikan[0]:Top speed pabrikan motor

Pabrikan[1]:299 km/jam

Pabrikan[2]:301 km/jam

Pabrikan[3]:299 km/jam

Pabrikan[4]:329 km/jam

BUILD SUCCESSFUL (total time: 0 seconds)
```

TUGAS INDIVIDU

1. Apa perbedaan Inheritance dengan Polimorphism,
Jelaskan Inheritance (Pewarisan)

Inheritance (penurunan sifat / pewarisan), ini merupakan ciri khas dari OOP yang tidak terdapat pada pemrograman procedural gaya lama. Dalam hal ini, inheritance bertujuan membentuk obyek baru yang memiliki sifat sama atau mirip dengan obyek yang sudah ada sebelumnya (pewarisan)

Sedangkan

Polymorphism, suatu aksi yang memungkinkan pemrogram menyampaikan pesan tertentu keluar dari hirarki obyeknya, dimana obyek yang berbeda memberikan tanggapan / respon terhadap pesan yang sama sesuai dengan sifat masing-masing obyek.

Atau Polymorphism dapat berarti banyak bentuk, maksudnya yaitu dapat menimpa (override), suatu method, yang dimana object tersebut diturunkan, sehingga memiliki kelakuan yang berbeda.

4. Kesimpulan

Buatlah kesimpulan dari pembahasan praktikum yang telah Anda kerjakan.

Pada praktikum modul 6 ini membahas tentang polymorphism dalam program java. yaitu memiliki nama sama, tetapi memiliki kelakuan (behavior) yang berbeda.

5. Referensi

Pada bagian ini cantumkan daftar pustaka/referensi yang ada gunakan dalam menyusun laporan praktikum ini.

Modul praktikum