

**LAPORAN PRAKTIKUM**  
**PEMROGRAMAN BERORIENTASI OBJEK**

**(Dosen Pengampu: *Dede Husen, M.Kom*)**



**NAMA : MUHAMMAD RIZAL NURFIRDAUS**

**NIM : 20230810088**

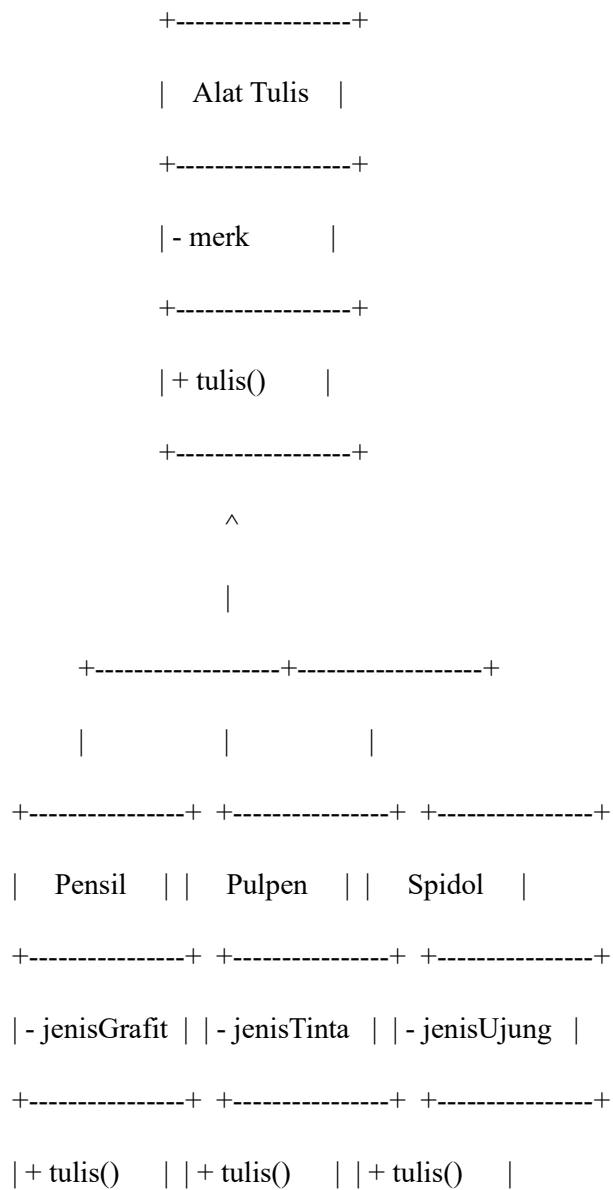
**KELAS : TINFC-2023-04**

**TEKNIK INFORMATIKA**  
**FAKULTAS ILMU KOMPUTER**  
**UNIVERSITAS KUNINGAN**

**2024**

## PRETEST

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



## PRAKTIKUM 1

```
package com.example;
```

```
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");}

    public void hapus(){System.out.println("Hapus Lingkaran\n");}

}

class Elips extends Bentuk{

    public void gambar(){System.out.println("Gambar Elips");}

    public void hapus(){System.out.println("Hapus Elips\n");}

}

class Segitiga extends Bentuk{

    public void gambar(){System.out.println("Gambar Segitiga");}

    public void hapus(){System.out.println("Hapus Segitiga\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();

    }

}

```



```

PS C:\bpo> & 'C:\Program Files\Eclipse Adoptium\jdk-21.0.3.9-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-c
p' 'C:\bpo\rizal\target\classes' 'com.example.Tesbentuk'
Bentuk
Menggambar
Menghapus gambar

Bentuk Lingkaran
Gambar Lingkaran
Hapus Lingkaran

Bentuk Elips
Gambar Elips
Hapus Elips

Bentuk Segitiga
Gambar Segitiga
Hapus Segitiga

PS C:\bpo>

```

## PRAKTIKUM 2

```
package com.example;
```

```

class Bentukwajah {

    public String respons()

        {return ("Perhatikan reaksi wajah saya\n");}

}

```

```

class Senyum extends Bentukwajah{

    public String respons()

        {return("Senyum karena senang\n");}

}

```

```

class Tertawa extends Bentukwajah{

    public String respons()

        {return("Tewtawa karena gembira\n");}

}

```

```

}

class Marah extends Bentukwajah{

    public String respons()

    {return("Kemarahan disebabkan bertengkar\n");}

}

class Sedih extends Bentukwajah{

    public String respons()

    {return("Sedih disebabkan cemburu\n");}

}

package com.example;

public class Programpolimorphism {

    public static void main(String[] args) {

        System.out.println("\nProgram Polimorphis");

        System.out.println("=====");

        Bentukwajah objbentuk=new Bentukwajah();

        Senyum objsenyum=new Senyum();

        Tertawa objtertawa=new Tertawa();

        Marah objmarah=new Marah();

        Sedih objsedih=new Sedih();

        Bentukwajah []Bentuk=new Bentukwajah[5];

        Bentuk[0]=objbentuk;

        Bentuk[1]=objsenyum;

        Bentuk[2]=objtertawa;

        Bentuk[3]=objmarah;

```

```

        Bentuk[4]=objsedih;

        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());

    }

}

```

```

PS C:\bpo> & 'C:\Program Files\Eclipse Adoptium\jdk-21.0.3.9-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\bpo\rizal\target\classes' 'com.example.Programpolimorphism'

Program Polimorphis
=====
Bentuk[0]:Perhatikan reaksi wajah saya

Bentuk[1]:Senyum karena senang

Bentuk[2]:Tewtawa karena gembira

Bentuk[3]:Kemarahan disebabkan bertengkar

Bentuk[4]:Sedih disebabkan cemburu

PS C:\bpo> 

```

## PRAKTIKUM 3

```
package com.example;
```

```
public class Hewan
```

```
{String jenis,ciri;
```

```
    public void suara(){System.out.println("Suara Hewan");}
```

```
    public void berjalan(){System.out.println("Cara Berjalan Hewan");}
```

```
}
```

```
class Singa extends Hewan{
```

```
    public void suara(){System.out.println("WaouWaou");}
```

```

public void berjalan(){

    System.out.println("Berlari");

    System.out.println("=====");

}

}

class Elang extends Hewan{

    public void suara(){System.out.println("Leooooookkk");}

    public void berjalan(){

        System.out.println("Terbang");

        System.out.println("=====");

    }

}

class Lebah extends Hewan{

    public void suara(){System.out.println("Wunngggggg");}

    public void berjalan(){

        System.out.println("Terbang");

        System.out.println("=====");

    }

}

class Paus extends Hewan{

    public void suara(){System.out.println("Leooooookkk");}

    public void berjalan(){

        System.out.println("Berenang");

        System.out.println("=====");

    }

}

```

```

    }

package com.example;

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("Paus");

        h = HewanPaus; h.suara(); h.berjalan();

        System.out.println("Singa");

        h = HewanSinga; h.suara(); h.berjalan();

        System.out.println("Elang");

        h = HewanElang; h.suara(); h.berjalan();

    }

}

```





```
PS C:\bpo> & 'C:\Program Files\Eclipse Adoptium\jdk-21.0.3-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-c
p' 'C:\bpo\rizal\target\classes' 'com.example.DemoHewan'
Lebah
Wunngggggg
Terbang
=====
Paus
Leooookkk
Berenang
=====
Singa
WaouWaou
Berlari
=====
Elang
Leooookkk
Terbang
=====
PS C:\bpo>
```

## POSTTEST

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

```
package com.example.m6;
```

```
public class Kendaraan {
```

```
    String jenis, ciri;
```

```
    public void suara() {
```

```
        System.out.println("Suara Kendaraan");
```

```
    }
```

```
    public void berjalan() {
```

```
        System.out.println("Cara Berjalan Kendaraan");
```

```
    }
```

```
}
```

```
class Motor extends Kendaraan {
```

```
    public void suara() {
```

```
        System.out.println("Brrmmm Brrmmm");
```

```
    }
```

```

public void berjalan() {

    System.out.println("Mengendarai di Jalan Raya");

    System.out.println("=====");

}

}

class Mobil extends Kendaraan {

    public void suara() {

        System.out.println("Stututututu Stututututu");

    }

    public void berjalan() {

        System.out.println("Mengemudi di Jalan Raya");

        System.out.println("=====");

    }

}

class Perahu extends Kendaraan {

    public void suara() {

        System.out.println("Tuut Tuut");

    }

    public void berjalan() {

        System.out.println("Berlayar di Air");

        System.out.println("=====");

    }

}

class Pesawat extends Kendaraan {

    public void suara() {

```

```

        System.out.println("Whooosh");
    }

    public void berjalan() {

        System.out.println("Terbang di Udara");

        System.out.println("=====");

    }

}

package com.example.m6;

public class DemoKendaraan {

    public static void main(String[] args) {

        Motor kendaraanMotor = new Motor();

        Mobil kendaraanMobil = new Mobil();

        Perahu kendaraanPerahu = new Perahu();

        Pesawat kendaraanPesawat = new Pesawat();

        Kendaraan k;

        System.out.println("Motor");

        k = kendaraanMotor; k.suara(); k.berjalan();

        System.out.println("Mobil Nissan Skyline");

        k = kendaraanMobil; k.suara(); k.berjalan();

        System.out.println("Perahu");

        k = kendaraanPerahu; k.suara(); k.berjalan();

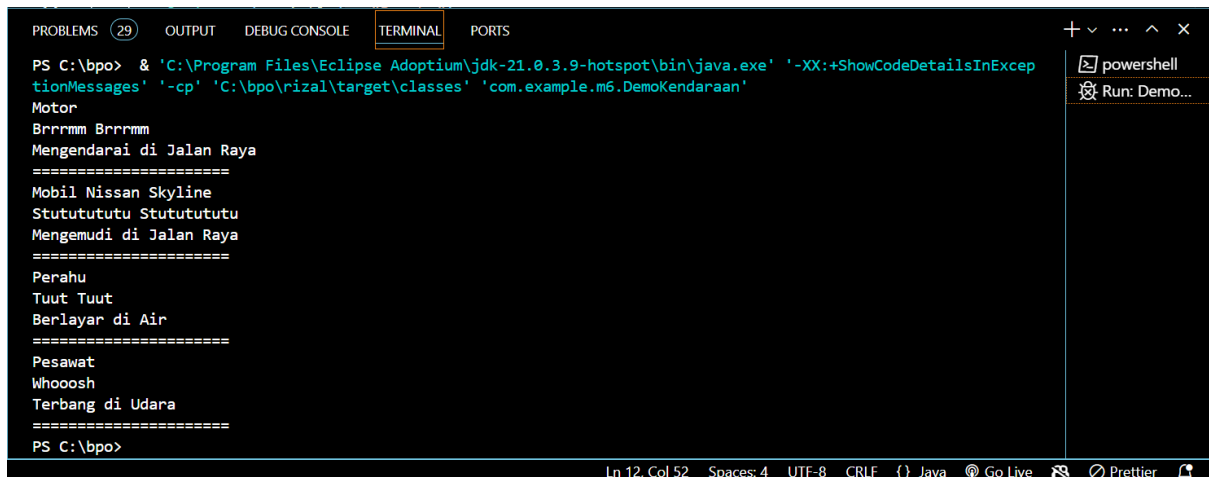
        System.out.println("Pesawat");

        k = kendaraanPesawat; k.suara(); k.berjalan();

    }
}

```

}



The screenshot shows the Eclipse IDE's integrated terminal. The terminal window has tabs for PROBLEMS (29), OUTPUT, DEBUG CONSOLE, TERMINAL (selected), and PORTS. The terminal content shows a PowerShell prompt where a Java command is executed. The output of the program is as follows:

```
PS C:\bpo> & 'C:\Program Files\Eclipse Adoptium\jdk-21.0.3.9-hotspot\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\bpo\rizal\target\classes' 'com.example.m6.DemoKendaraan'
Motor
Brrrrm Brrrrm
Mengendarai di Jalan Raya
=====
Mobil Nissan Skyline
Stututututu Stututututu
Mengemudi di Jalan Raya
=====
Perahu
Tuut Tuut
Berlayar di Air
=====
Pesawat
Whooosh
Terbang di Udara
=====
PS C:\bpo>
```

On the right side of the terminal, there is a 'powershell' icon and a 'Run: Demo...' button. The status bar at the bottom indicates 'Ln 12, Col 52', 'Spaces: 4', 'UTF-8', 'CRLF', and the active language is 'Java'. Other icons for 'Go Live' and 'Prettier' are also visible.

## TUGAS INDIVIDU UNTUK MINGGU DEPAN

1. Apa perbedaan Inheritance dengan Polimorphism, Jelaskan!

Inheritance (pewarisan) dan polymorphism (polimorfisme) adalah konsep dasar dalam pemrograman berorientasi objek. Inheritance memungkinkan sebuah kelas turunan untuk mewarisi atribut dan metode dari kelas induk, menciptakan hubungan "is-a" dan membantu dalam penggunaan kembali serta perluasan kode. Contohnya, kelas `Burung` mewarisi kelas `Hewan` dan menambahkan kemampuan khusus seperti `terbang()`. Polymorphism, di sisi lain, memungkinkan metode yang sama digunakan pada objek berbeda yang merespons sesuai dengan jenis mereka, memberikan fleksibilitas dalam pemrograman. Misalnya, metode `tulis()` yang diterapkan dalam berbagai kelas alat tulis seperti `Pensil`, `Pulpen`, dan `Spidol`, menghasilkan output yang berbeda tergantung pada objek yang memanggilnya.