

## PRAKTIKUM 4 PEWARISAN

### TUJUAN

- Mahasiswa memahami tentang pewarisan (inheritance)
- Mahasiswa memahami tentang polymorphisme
- Mahasiswa memahami tentang kelas abstrak

### DASAR TEORI

Pewarisan merupakan konsep dalam pemrograman berorientasi objek yang memungkinkan untuk membuat suatu kelas (subclass) dengan didasarkan pada kelas yang sudah ada (superclass) sehingga mewarisi semua metode dan variabelnya. Untuk melakukan ini, Java menyediakan keyword **extends** yang dapat dipakai saat mendeklarasikan class.

```
class alpha
{
    private int a=5;
    public void info( )
    {
        System.out.println("a="+a);
    }
}
class bravo extends alpha
{
    private int b=10;
    public void infolagi()
    {
        System.out.println("b="+b);
    }
}
public AlphaBravo
{
    public static void main(String args[])
    {
        alpha ob=new alpha();
        bravo obj=new bravo();
        ob.info();
        obj.info();
        obj.infolagi();
    }
}
```

Beberapa aturan tentang pewarisan yang perlu diperhatikan :

1. Java tidak mengizinkan pewarisan berganda, yaitu membuat suatu subclass yang diturunkan dari beberapa superclass sekaligus. Contoh : class Z hendak diturunkan dari class X dan juga sekaligus dari class Y. Hal yang memungkinkan dalam C++ tetapi tidak dalam Java.
2. Suatu subclass pada dasarnya adalah class biasa, maka tetap dapat melakukan pewarisan pada subclass ini.

Java menyediakan suatu mekanisme yang memungkinkan metode ditentukan dari dalam kelas tetapi tidak disertai dengan definisinya. Metode ini dikenal dengan nama metode Abstrak, sedangkan kelasnya dinamakan kelas abstrak.

```

public abstract class kendaraan
{
    protected String nama;
    public abstract void jalankan();
}
class mobil extends kendaraan
{
    public mobil(String nama)
    {
        this.nama=nama;
    }
    public void jalankan()
    {
        System.out.println("nama:"+this.nama+"cara menjalankan distater
mesinnya");
    }
}
public class JalankanMobil
{
    public static void main(String args[])
    {
        mobil mobilku=new mobil("sedan");
        mobilku.jalankan();
    }
}

```

Jika kita mempunyai variabel objek yang bertipe superkelas variabel objek ini bisa diisi dengan objek superkelas atau subkelas tanpa memerlukan perubahan tipe. Model ini dikenal dengan nama polimorfisme, yaitu kemampuan untuk berhubungan dengan berbagai tipe didasarkan pada fitur yang sama.

```

class makhluk
{
    public void info()
    {
        System.out.println("info makhluk");
    }
}
class mamalia extends makhluk
{
    public void info()
    {
        System.out.println("info mamalia");
    }
}
class sapi extends mamalia
{
    public void info()
    {
        System.out.println("info sapi");
    }
}

```

## **TUGAS PENDAHULUAN**

1. Keuntungan apa yang dapat diperoleh dengan melakukan pewarisan dan kapan kita mungkin perlu melakukannya!
2. Apa yang terjadi jika kita mendeklarasikan member pada subclass yang sama persis dengan yang dimiliki oleh superclassnya? Bagaimana pula cara subclass agar dapat mengakses member dari superclass tersebut?