



Riza Satria Perdana, S.T., M.T.

Teknik Informatika - STEI ITB

Polymorphism

Polymorphism

Pemrograman Berorientasi Objek



Polymorphism

- Disebut juga: Subtype polymorphism
- Subclass dapat mendefinisikan sifatnya sendiri dan dapat menggunakan fungsionalitas yang sama dengan parent class
- Kemampuan suatu objek memerankan banyak bentuk



Polymorphism ...

- Suatu entitas berkelakuan berbeda tergantung dari konteks yang digunakan
- Kemampuan melakukan aksi berbeda berdasarkan objek yang dikenai aksi tersebut

Polymorphisme di Java

- method overloading
- method overriding
- method overriding dengan cara meng-extend sebuah class

Contoh

```
public void printDescription() {  
    System.out.println("\nBike is " + "in gear " + this.gear + " with a cadence of "  
        + this.cadence + " and travelling at a speed of " + this.speed + ". ");  
}
```

```
public class MountainBike extends Bicycle {  
    private String suspension;  
    public MountainBike(int startCadence, int startSpeed, int startGear, String suspensionType) {  
        super(startCadence, startSpeed, startGear);  
        this.setSuspension(suspensionType);  
    }  
    public String getSuspension() {  
        return this.suspension;  
    }  
    public void setSuspension(String suspensionType) {  
        this.suspension = suspensionType;  
    }  
    public void printDescription() {  
        super.printDescription();  
        System.out.println("The " + "MountainBike has a" + getSuspension() + " suspension.");  
    }  
}
```



Contoh ...

```
public class RoadBike extends Bicycle{
    private int tireWidth; // In millimeters (mm)

    public RoadBike(int startCadence, int startSpeed, int startGear, int newTireWidth) {
        super(startCadence, startSpeed, startGear);
        this.setTireWidth(newTireWidth);
    }

    public int getTireWidth() {
        return this.tireWidth;
    }

    public void setTireWidth(int newTireWidth) {
        this.tireWidth = newTireWidth;
    }

    public void printDescription() {
        super.printDescription();
        System.out.println("The RoadBike" + " has " + getTireWidth() + " MM tires.");
    }
}
```



Contoh ...

```
public class TestBikes {  
    public static void main(String[] args) {  
        Bicycle bike01, bike02, bike03;  
  
        bike01 = new Bicycle(20, 10, 1);  
        bike02 = new MountainBike(20, 10, 5, "Dual");  
        bike03 = new RoadBike(40, 20, 8, 23);  
  
        bike01.printDescription();  
        bike02.printDescription();  
        bike03.printDescription();  
    }  
}
```



Terima Kasih