

Bahasa Pemrograman :: Polimorfism and Abstraction

Julio Adisantoso
ILKOM IPB

24 Mei 2010

Access Modifiers

- **Akses Default** (Package Accessibility)
Tidak terdapat keyword pada tipe ini. Hanya class dalam package yang sama yang memiliki hak akses terhadap variabel dan methods dalam class.
- **Akses Public**
Mengijinkan seluruh class member untuk diakses baik dari dalam dan luar class.
- **Akses Protected**
Hanya mengijinkan class member untuk diakses oleh method dalam class tersebut dan elemen-elemen subclass.
- **Akses Private**
Pengaksesan class hanya dapat diakses oleh class dimana tipe ini dibuat.

Accessor Methods

- Method yang dibuat agar object lain dapat mengakses data private. Dalam hal ini kita gunakan accessor methods.
- Accessor Methods digunakan untuk membaca nilai variabel pada class, baik berupa instance maupun static.
- Contoh:

```
public class Person {  
    protected String name;  
    ...  
    public String getName() {  
        return name;  
    }  
}
```

Mutator Methods

- Method yang dapat memberi atau mengubah nilai variable dalam class, baik itu berupa instance maupun static.
- Contoh:

```
public class Person {  
    protected String name;  
    ...  
    public void setName( String name ) {  
        this.name = name;  
    }  
}
```

Inheritance

- Mekanisme pewarisan, artinya beberapa sifat super class diturunkan/diwariskan ke sub-class. Contoh: class Person dan Student.
- Contoh:

```
public class Student extends Person {  
    ...  
    ...  
    ...  
}
```

Polimorfism

Perhatikan class berikut:

```
public class Person {
    public String getName(){
        System.out.println("Person Name:" + name);
        return name; }
}
public class Student extends Person {
    public String getName(){
        System.out.println("Student Name:" + name);
        return name; }
}
public class Employee extends Person {
    public String getName(){
        System.out.println("Employee Name:" + name);
        return name;
    }
}
```

Mengubah Sifat

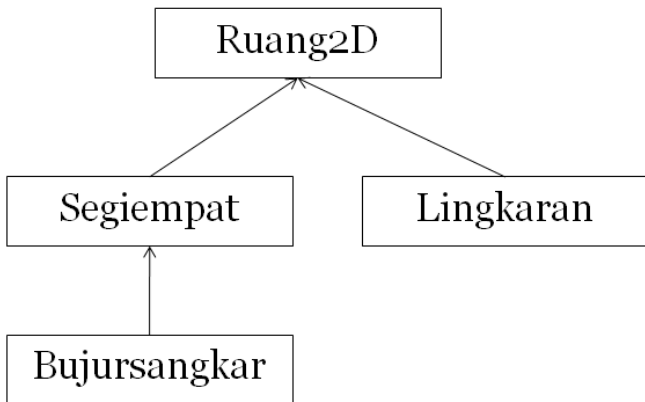
Perhatikan pemanggilan method getName() berikut:

```
public static void main( String[] args ) {  
    Person ref;  
    Student  s = new Student();  
    Employee e = new Employee();  
  
    ref = s; //Person menunjuk ke objek Student  
    String temp = ref.getName();  
    System.out.println( temp );  
  
    ref = e; //Person menunjuk ke objek Employee  
    temp = ref.getName();  
    System.out.println( temp );  
}
```

Pengertian

- Kemampuan dari referensi untuk mengubah sifat menurut objek apa yang dijadikan acuan dinamakan **polimorfisme**.
- Polimorfisme menyediakan *multiobject* dari *subclasses* yang berbeda untuk diperlakukan sebagai objek dari *superclass* tunggal, yang secara otomatis menunjuk *method* yang tepat untuk menggunakannya ke *particular object* berdasar *subclass* yang termasuk di dalamnya.

Abstract Class



Abstract Class

- Class Ruang2D mempunyai method **area()** yang akan di-override oleh subclasses nya.
- Ruang2D merupakan sifat umum dari suatu bidang dua dimensi (Segiempat, Lingkaran, Bujursangkar).
- Method dalam class Ruang2D **tidak memiliki implementasi**. Class jenis ini yang disebut dengan **abstract class**.
- Umumnya muncul pada hirarki class pemrograman berbasis object paling atas, dan mendefinisikan keseluruhan aksi yang mungkin pada object dari seluruh subclasses dalam class.

Abstract Method

- Method dalam abstract class yang tidak mempunyai implementasi dinamakan **abstract method**.
- Contoh:

```
public abstract class Ruang2D {  
    public abstract double area();  
}
```

Ruang tiga dimensi

Apa yang bisa dimanfaatkan dari class Ruang2D untuk membuat class Ruang3D berikut?

