

BAB 8

INTERFACE

Tujuan

1. Mampu memahami konsep Interface dalam pemrograman Java
2. Mampu mengimplementasikan konsep Interface dalam menyelesaikan permasalahan

Ringkasan Materi

A. Pengertian Interface

Pada bab sebelumnya telah dijelaskan bahwa polymorphism hanya memungkinkan untuk mewarisi sifat dari satu kelas abstrak saja, namun tidak jarang juga diperlukan untuk mewarisi sifat lebih dari satu class abstrak, atau biasa disebut juga dengan multiple inheritance. Dalam Java, tidak mendukung multiple inheritance, artinya sebuah class hanya bisa mewarisi sifat dari sebuah superclass atau yang biasa disebut dengan single inheritance. Sebuah Java class tersebut tidak mendukung prinsip multiple inheritance. Dalam memenuhi prinsip kebutuhan multiple inheritance, pada Java terdapat Interfaces yang memiliki aspek seperti multiple inheritance namun berbeda dengan abstract class. Sebuah interface hanya memiliki konstanta dan abstract method. Syntax untuk mendeklarasikan interface adalah sebagai berikut:

```
Modifier interface Interface_name {
    /** Konstanta */
    /** Abstract Method */
}
```

B. Implementasi Interface

Sebuah kelas dapat mengimplementasikan sebuah interface dengan menggunakan kata kunci *implements* sebagai berikut.

```
Modifier class class_name implements Interface_name {
    /** Attribute */
    /** Method */
}
```

Penggunaan kata kunci implements dapat lebih dari 2

```
Modifier      class      class_name      implements      Interface_name1,
interface_name2 {
    /** Attribute */
    /** Method */
}
```

Pelaksanaan Percobaan

A. Interface

Identitas.java	
1	package latihaninterface;
2	public interface Identitas {
3	public void tampilkanNama();
4	public void tampilkanUmur();
5	}

Manusia.java

```

1 package latihaninterface;
2 public class Manusia implements MakhlukHidup, Identitas {
3
4     private String nama;
5     private int umur;
6
7     @Override
8     public void makan() {
9         System.out.println("Makan pakai sendok garpu");}
10
11     @Override
12     public void berjalan() {
13         System.out.println("Jalan pakai dua kaki");}
14
15     @Override
16     public void bersuara() {
17         System.out.println("Suaranya merdu");}
18
19     @Override
20     public void tampilkanNama() {
21         System.out.println("Nama saya: " + this.nama);}
22
23     @Override
24     public void tampilkanUmur() {
25         System.out.println("Umur saya: " + this.umur);}
26
27 }

```

Hewan.java

```

1 package latihaninterface;
2 public class Hewan implements MakhlukHidup, Identitas {
3
4     @Override
5     public void makan() {
6         System.out.println("Makan pakai tangan dan mulut");
7     }
8
9     @Override
10    public void berjalan() {
11        System.out.println("Jalan pakai 4 kaki");
12    }
13
14    @Override
15    public void bersuara() {
16        System.out.println("Suaranya nggak jelas");
17    }
18
19    public void tampilkanNama (){}
20
21    public void tampilkanUmur () {}
22 }

```

MakhlukHidup.java

```

1 package latihaninterface;
2 public interface MakhlukHidup {
3     public void makan();
4     public void berjalan();
5     public void bersuara();
6 }

```

Data dan Analisis hasil percobaan

Pertanyaan

```

1 package praktikumpl;
2
3 public interface Colorable {
4     public void howToColor();
5 }
6 public interface Comparable
7
8 {
9     public void compareTo(Object obj);
10 }
11 public class Rectangle implements Colorable, Comparable{ // lass
12     rectanggle
13     private String warna;
14     private int kategori;
15
16     public Rectangle() {
17     }
18
19     public Rectangle(String warna) {
20         this.warna = warna;
21     }
22
23     public void howToColor() {
24         if(this.warna == null){
25             System.out.println("tidak ada warna, warna bangun kotak
26 masih polos");
27         }
28         else{
29             System.out.println("bangun kotak sudah diwarnai dengan
30 warna "+this.warna);
31         }
32     }
33
34     public void compareTo(Object obj) {
35         this.kategori = (int) obj;
36         if(this.kategori == 0){
37             System.out.println("ukuran cat yang cocok untuk bangun
38 kotak dengan ukuran kategori " +this.kategori+" yaitu 2.5L" );
39         }
40         else{
41             System.out.println("ukuran cat yang cocok untuk bangun kotak
42 dengan ukuran kategori " +this.kategori+" yaitu 6.5L" );
43         }
44     }
45 }
46 public static void main(String[] args) {
47
48     Rectangle kotak1 = new Rectangle("merah");
49     Rectangle kotak2= new Rectangle();
50     Rectangle kotak3 = new Rectangle();
51     kotak1.howToColor();
52     kotak2.howToColor();
53     kotak3.compareTo(4);
54
55 }

```

1. Lakukan percobaan diatas dan benahi jika menemukan kesalahan serta jelaskan!

Tidak terdapat kesalahan dalam kode program diatas

```

project_root > latihaninterface > J Manusia.java > Main
3 public class Manusia implements MakhlukHidup, Identitas {
27
28     @Override
29     public void tampilkanNama() {
30         System.out.println("Nama saya: " + this.nama);
31     }
32
33     @Override
34     public void tampilkanUmur() {
35         System.out.println("Umur saya: " + this.umur);
36     }
37 }
38
39 class Main {
    Run | Debug
40     public static void main(String[] args) {
41         Manusia saya = new Manusia(nama:"Lia", umur:20);
42         saya.makan();
43         saya.berjalan();
44         saya.bersuara();
45         saya.tampilkanNama();
46         saya.tampilkanUmur();
47     }
48 }

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

Run: Main + - □ □ □ ... ^ x

```

Makan pakai sendok garpu
Jalan pakai dua kaki
Suaranya merdu
Nama saya: Lia
Umur saya: 20
PS C:\Users\HPA\Documents>Tugas 1 PBO\Modul 9>

```

7. Ubah source code diatas menjadi proses meminta inputan dari user dan buat menjadi interaktif!

```

project_root > latihaninterface > J Main.java > Main
1 package project_root.latihaninterface;
2
3 import java.util.Scanner;
4
5 public class Main {
6     public static void main(String[] args) {
7         Scanner scanner = new Scanner(System.in);
8
9         System.out.print("Masukkan nama: ");
10        String nama = scanner.nextLine();
11
12        System.out.print("Masukkan umur: ");
13        int umur = scanner.nextInt();
14
15        Manusia saya = new Manusia(nama, umur);
16
17        System.out.println("Informasi tentang diri saya:");
18        saya.makan();
19        saya.berjalan();
20        saya.bersuara();
21        saya.tampilkanNama();
22        saya.tampilkanUmur();
23    }
24 }

```

OUTPUT

```

Masukkan nama: Lia
Masukkan umur: 20
Informasi tentang diri saya:
Makan pakai sendok garpu
Jalan pakai dua kaki
Suaranya merdu
Nama saya: Lia
Umur saya: 20
PS C:\Users\VIP\Documents\Tugas 1_PBO\Modul 9>

```

8. Buat objek selain objek diatas dengan menggunakan method yang berbeda dengan yang diatas! (min.1 contoh)

```

project_root > latihaninterface > J Hewan.java > Hewan
3 public class Hewan implements MakhlukHidup, Identitas {
4     private String nama;
5     private int umur;
6
7     public Hewan(String nama, int umur) {
8         this.nama = nama;
9         this.umur = umur;
10    }
11
12    @Override
13    public void makan() {
14        System.out.println("Makan pakai tangan dan mulut");
15    }
16
17    @Override
18    public void berjalan() {
19        System.out.println("Jalan pakai 2 kaki");
20    }
21
22    @Override
23    public void bersuara() {
24        System.out.println("Suaranya nggak jelas");
25    }
26
27    @Override
28    public void tampilkanNama() {
29        System.out.println("Nama saya: " + this.nama);
30    }
31
32    @Override
33    public void tampilkanUmur() {
34        System.out.println("Umur saya: " + this.umur);
35    }
36 }

```

```

project_root > latihaninterface > J Main.java > ...
1 package project_root.latihaninterface;
2 import java.util.Scanner;
3
4 public class Main {
5     public static void main(String[] args) {
6         Scanner scanner = new Scanner(System.in);
7
8         System.out.print("Enter the name: ");
9         String name = scanner.nextLine();
10
11        System.out.print("Enter the age: ");
12        int age = scanner.nextInt();
13
14        Hewan hewan = new Hewan(name, age);
15
16        hewan.makan();
17        hewan.berjalan();
18        hewan.bersuara();
19        hewan.tampilkanNama();
20        hewan.tampilkanUmur();
21    }
22 }
23

```

OUTPUT

```
Enter the name: monyet
Enter the age: 4
Makan pakai tangan dan mulut
Jalan pakai 2 kaki
Suaranya nggak jelas
Nama saya: monyet
Usur saya: 4
PS C:\Users\HP\Documents\Tugas_1_PBO\Modul 9> |
```

PENJELASAN SINGKAT :

Kode program diatas meminta user untuk memasukkan nama dan usia, membuat Hewan objek dengan nilai tersebut, dan kemudian memanggil metodenya untuk menampilkan property dan perilakunya.

Tugas Praktikum

Perusahaan NV. Meneer memiliki koperasi karyawan yang memungkinkan karyawannya berbelanja di koperasi tersebut. Tentunya, karyawan tersebut bisa membayar belanjanya tersebut di akhir bulan melalui pemotongan gaji. Ada 2 kelas yang terlibat disini, Invoice dan Employee. Kedua class tadi mengimplementasikan **interface Payable** yang mana ia hanya memiliki satu method yang harus diimplementasikan di kedua class, yaitu **getPayableAmount()**. Program harus bisa mengolah gaji karyawan di akhir bulan beserta invoice belanjaan karyawan yang nantinya gaji karyawan perbulannya dikurang total harga belanjanya secara polimorfis. Tampilkan informasi dari karyawan tersebut beserta total gaji setelah dipotong hutang belanjaan di koperasi dan tampilkan pula detail belanjanya secara polimorfis pula.

1. Atribut dari Invoice:

String productName, Integer quantity, Integer pricePerItem

2. Atribut dari Employee:

Integer registrationNumber, String name, Integer salaryPerMonth, Invoice[] invoices