

## MODUL 9 – Abstraksi dan Kelas Abstrak

### 9.1. CAPAIAN PEMBELAJARAN

- 1 Mahasiswa mampu menjelaskan konsep abstraksi dan kelas abstrak dalam Pemrograman Berorientasi Objek (PBO).
- 2 Mahasiswa mampu mengimplementasikan kelas abstrak dalam PBO.

### 9.2. ALAT DAN BAHAN

- 1 Seperangkat komputer lengkap/Laptop dengan koneksi internet
- 2 Web Browser (Chrome/Firefox/Opera/Edge/Safari/dll)
- 3 Aplikasi Kantor (Microsoft Office/Libre Office/WPS Office/etc)
- 4 JDK (<https://www.oracle.com/java/technologies/downloads/>)
- 5 Netbeans (<https://netbeans.apache.org/front/main/download/>)

### 9.3. DASAR TEORI

#### 9.3.1. Abstraksi

Abstraksi adalah salah satu pilar utama dalam Pemrograman Berorientasi Objek (PBO). Tujuannya adalah untuk menyembunyikan detail implementasi dan menampilkan hanya fungsionalitas penting dari suatu objek. Abstraksi membantu dalam memecah sistem yang kompleks menjadi bagian-bagian yang lebih sederhana.

Dengan abstraksi, kita fokus pada apa yang dilakukan objek, bukan bagaimana caranya.

Contoh konsep abstraksi: Misalnya, pada objek kendaraan, kita hanya perlu tahu bahwa kendaraan tersebut bisa berjalan, tetapi kita tidak perlu tahu bagaimana mesin internalnya bekerja.

#### 9.3.2. Kelas Abstrak

Kelas abstrak adalah kelas yang tidak dapat diinstansiasi langsung. Kelas ini digunakan sebagai cetak biru bagi kelas lain yang mewarisinya. Kelas abstrak bisa memiliki metode abstrak (metode tanpa implementasi) dan metode biasa (metode dengan implementasi).

Karakteristik Kelas Abstrak:

- Tidak dapat dibuat objeknya secara langsung.
- Bisa memiliki metode abstrak dan metode dengan implementasi.
- Subclass yang mengimplementasikan kelas abstrak wajib

mengimplementasikan semua metode abstrak yang ada.

## 9. 4. PRAKTIKUM

### 6.4.1. Perisapan

- 1) Buka Netbeans yang sudah terinstall pada komputer
- 2) Buat proyek baru dengan nama PraktikumPBO\_9.
- 3) Buat package baru dengan nama praktikum9.

### 6.4.2. Studi Kasus: Kendaraan

- 1) Buat kelas Abstrak kendaraan dengan kode program sebagai berikut :

```
// Definisi kelas abstrak
abstract class Kendaraan {
    // Metode abstrak
    abstract void berjalan();

    // Metode dengan implementasi
    void info() {
        System.out.println("Ini adalah kendaraan.");
    }
}
```

- 2) Buat kelas turunan dengan kode sebagai berikut :

```
// Kelas turunan dari Kendaraan
class Mobil extends Kendaraan {
    @Override
    void berjalan() {
        System.out.println("Mobil sedang berjalan
dengan cepat.");
    }
}

class Sepeda extends Kendaraan {
    @Override
    void berjalan() {
        System.out.println("Sepeda sedang berjalan
dengan pelan.");
    }
}
```

- 3) Buat main class dengan kode sebagai berikut :

```
// Kode utama untuk pengujian
public class Main {
    public static void main(String[] args) {
        Kendaraan mobil = new Mobil();
        Kendaraan sepeda = new Sepeda();
    }
}
```

```
        mobil.berjalan();  
        mobil.info();  
  
        sepeda.berjalan();  
        sepeda.info();  
    }  
}
```

4) Cobalah untuk membuat objek lain dari kelas kendaraan, apakah bisa?

## 9.5. TUGAS MODUL 9

### 9.5.1. Soal

- Buat kelas abstrak Hewan dengan metode abstrak suara().
- Buat dua kelas turunan, yaitu Kucing dan Anjing, yang mengimplementasikan metode suara() sesuai dengan suara hewan tersebut.
- Contoh Output yang Diharapkan:
- Kucing mengeluarkan suara: Meong
- Anjing mengeluarkan suara: Guk Guk

### 9.5.2. Petunjuk Pengerjaan

#### a) Laporan:

- Buatlah laporan akhir sesuai dengan praktikum yang dilakukan
- Tuliskan laporan ke dalam akun masing-masing web medium
- Publikasikan laporan tersebut
- Kirimkan tautan laporan ke elita.
- **Batas Pengumpulan:** Sebelum Pertemuan Praktik Ke 10.