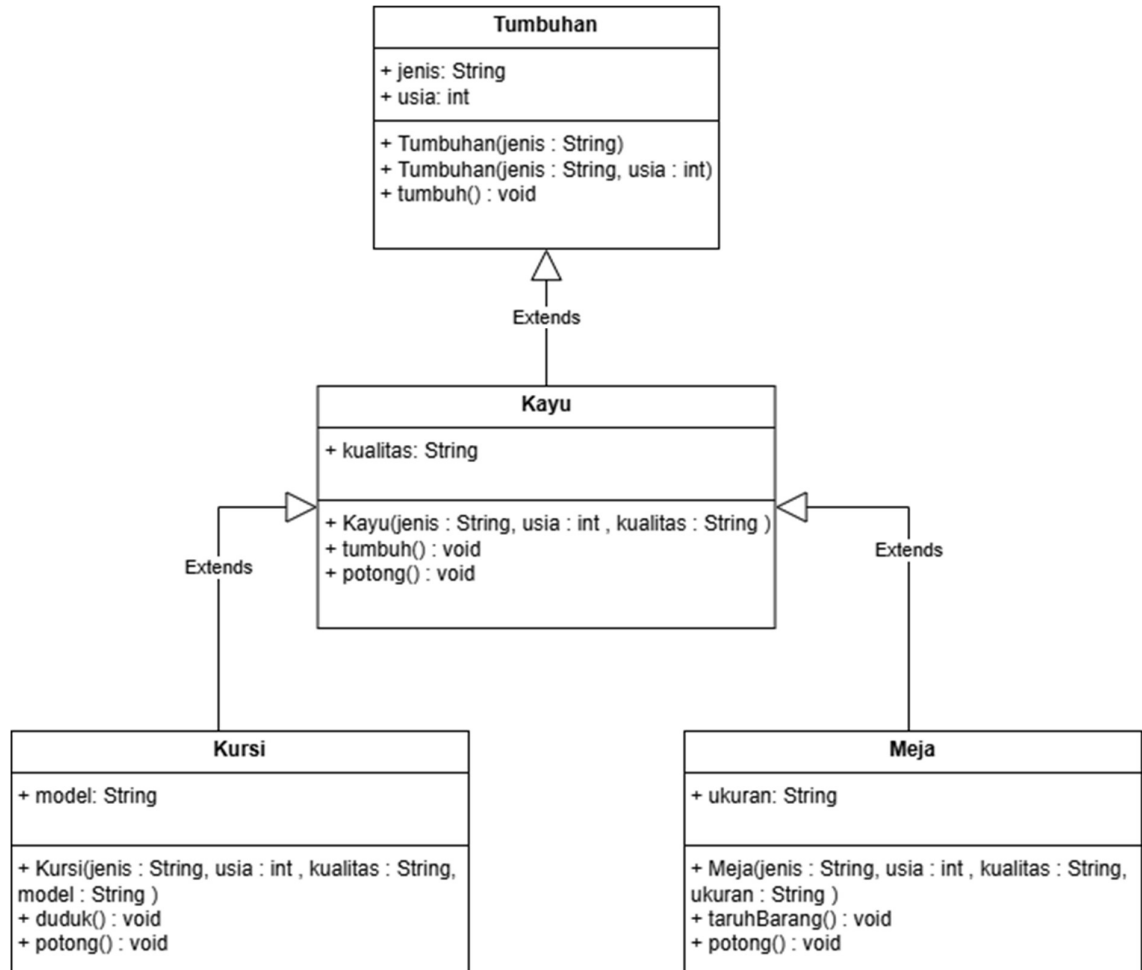


## Overloading Overriding

Oleh : Vincentius Leonanda Prabowo

2341720149 / 28

### 1. UML



## 2. CODE

```
class Tumbuhan {
    String jenis;
    int usia;

    // Constructor Overloading
    public Tumbuhan(String jenis) {
        this.jenis = jenis;
        this.usia = 0; // default usia
    }

    public Tumbuhan(String jenis, int usia) {
        this.jenis = jenis;
        this.usia = usia;
    }

    // Method Overriding Example
    public void tumbuh() {
        System.out.println(jenis + " sedang tumbuh.");
    }

    // Method Overloading
    public void tumbuh(int tahun) {
        this.usia += tahun;
        System.out.println(jenis + " tumbuh selama " + tahun + " tahun.");
    }
}
```

```
class Kayu extends Tumbuhan {
    String kualitas;

    // Constructor Overloading
    public Kayu(String jenis, int usia, String kualitas) {
        super(jenis, usia); // panggil constructor superclass
        this.kualitas = kualitas;
    }

    // Method Overriding
    @Override
    public void tumbuh() {
        System.out.println("Kayu berkualitas " + kualitas + " dari jenis " + jenis + " sedang tumbuh.");
    }

    public void potong() {
        System.out.println("Kayu dari " + jenis + " dipotong.");
    }
}
```

```

class Kursi extends Kayu {
    String model;

    // Constructor
    public Kursi(String jenis, int usia, String kualitas, String model) {
        super(jenis, usia, kualitas);
        this.model = model;
    }

    // Method Overriding
    @Override
    public void potong() {
        System.out.println("Kayu diolah menjadi kursi model " + model);
    }

    public void duduk() {
        System.out.println("Kursi model " + model + " digunakan untuk duduk.");
    }
}

```

```

class Meja extends Kayu {
    String ukuran;

    // Constructor
    public Meja(String jenis, int usia, String kualitas, String ukuran) {
        super(jenis, usia, kualitas);
        this.ukuran = ukuran;
    }

    // Method Overriding
    @Override
    public void potong() {
        System.out.println("Kayu diolah menjadi meja ukuran " + ukuran);
    }

    public void taruhBarang() {
        System.out.println("Meja ukuran " + ukuran + " digunakan untuk menaruh barang.");
    }
}

```

```
public class OverloadingOverriding {  
    public static void main(String[] args) {  
        // Tumbuhan dengan Overloading Constructor  
        Tumbuhan tumbuhan = new Tumbuhan("Pohon Jati", 5);  
        tumbuhan.tumbuh(); // Overriding method dari superclass Tumbuhan  
        tumbuhan.tumbuh(3); // Overloading method dengan parameter tahun  
  
        // Kayu dengan Overriding Method  
        Kayu kayu = new Kayu("Pohon Mahoni", 10, "Tinggi");  
        kayu.tumbuh(); // Overriding method di Kayu  
        kayu.potong(); // Memanggil method potong() dari Kayu  
  
        // Kursi dengan Overriding Method  
        Kursi kursi = new Kursi("Pohon Oak", 8, "Sedang", "Minimalis");  
        kursi.tumbuh(); // Overriding method di Kayu  
        kursi.potong(); // Overriding method di Kursi  
        kursi.duduk(); // Method dari kelas Kursi  
  
        // Meja dengan Overriding Method  
        Meja meja = new Meja("Pohon Pinus", 6, "Rendah", "Besar");  
        meja.tumbuh(); // Overriding method di Kayu  
        meja.potong(); // Overriding method di Meja  
        meja.taruhBarang(); // Method dari kelas Meja  
    }  
}
```