

Kelas Abstrak

Abstrak merupakan bentuk khusus dari inheritance atau pewarisan yang memungkinkan adanya keharusan untuk menuliskan kembali method dari class induk. Class yang memiliki sifat tersebut disebut Kelas Abstrak atau Abstract Class. Kelas Abstrak memiliki abstract method (berupa prosedur dan fungsi), yang harus dituliskan kembali oleh class yang meng-extends abstract class tersebut.

Kelas abstrak memiliki sifat tidak dapat diinstansiasi dan digunakan hanya untuk diturunkan ke dalam bentuk kelas konkret atau kelas abstrak berikutnya. Dalam class diagram, abstract class dan abstract method dituliskan menggunakan huruf miring.

Untuk pengkodean, kelas abstrak dideklarasikan menggunakan keyword abstract. Setiap abstract method tidak memiliki body method, sehingga pada pendeklarasian diakhiri titik koma (;), bukan kurung kerawal buka-tutup ({}).

Bentuk umum:

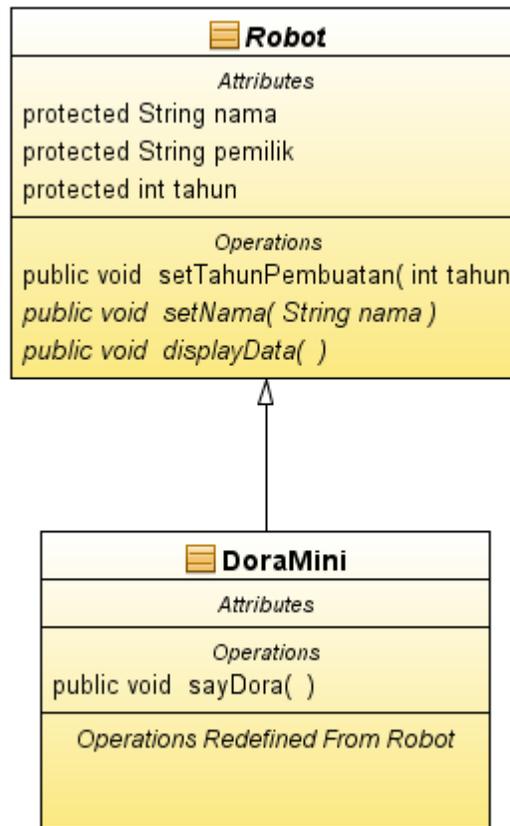
```
public abstract class NamaKelasAbstrak{
    /*deklarasi atribut dan konstanta*/
    public abstract void namaProsedurAbstrak();
    public abstract int namaFungsiAbstrak();
    public abstract String namaFungsiAbstrakDua();
    /* kode lainnya*/
}
```

Contoh Kasus:

Terdapat sebuah class diagram sebagai berikut.

Buatlah 2 kelas berdasarkan diagram kelas di atas (beserta kelas Main). Keterangan tambahan:

- Kelas Robot adalah kelas abstrak
- Method “sayDora” akan menampilkan “Halo, Saya Dora Mini” di layar
- Method displayData akan menampilkan setiap nilai dari atribut yang dimiliki ke layar
- Method setNama dan displayData adalah method abstract



Kode:

```

/*Robot.java*/

public abstract class Robot{
    protected String nama;
    protected String pemilik;
    protected int tahun;

    public void setTahunPembuatan(int tahun){
        this.tahun = tahun;
    }
    public abstract void setNama(String nama);
    public abstract void displayData();
}
  
```

```

/*DoraMiniAbs.java*/

public class DoraMiniAbs extends Robot{
    public void sayDora(){
        System.out.println("Halo, Saya Dora Mini");
    }
    public void setName(String nama){
        super.nama = nama;
    }
    public void displayData(){
        System.out.println("Nama: "+super.nama);
        System.out.println("Pemilik: "+super.pemilik);
        System.out.println("Tahun Produksi: "+super.tahun);
    }
}

```

```

/*Main.java*/

public class MainAbstrak{
    public static void main(String args[]){
        //Robot r = new Robot();
        DoraMini dm = new DoraMini();
        dm.setName("Dora Mini Satu");
        dm.setTahunPembuatan(2014);
        dm.displayData();
        dm.sayDora();
    }
}

```

Perhatikan bahwa:

1. Abstract method hanya terdapat pada abstract class, tapi abstract class boleh tidak mengandung abstract method.
2. DoraMini meng-override semua method abstract dari class Robot.
3. Instansiasi dari class Robot (comment code pada Main.java), tidak mungkin dilakukan, karena class bersifat abstract.

Hasil:

```
[rbudiawan@eja Java]$ javac MainAbstrak.java
[rbudiawan@eja Java]$ java MainAbstrak
Nama:Dora Mini Satu
Pemilik:null
Tahun Produksi:2014
Say Dora
[rbudiawan@eja Java]$ █
```

Note: Setiap nama class seharusnya menyesuaikan dengan nama class diagram. Perbedaan nama terjadi karena beberapa alasan.