

Classes and Objects

Class

- Class pada dasarnya adalah sebuah template atau blueprint dari object
- Sebuah class tidak hanya dapat digunakan untuk membuat sebuah object saja, tapi beberapa object
- Instansiasi merupakan langkah untuk membuat sebuah object dari sebuah class dengan cara memanggil constructor pada class tersebut (Penjelasan mengenai constructor akan dibahas lebih lanjut pada pertemuan berikutnya)

Class

Example

```
class MyClass {  
    // code goes here  
}
```

- Penamaan class bersifat case sensitive
- Dalam penamaan sebuah class, sebaiknya menggunakan camel case, seperti pada contoh di atas, class tersebut diawali dengan huruf kapital, pada kata kedua juga menggunakan huruf kapital

Object

- Object biasanya memiliki state dan behavior
- Contoh object mobil memiliki state (merek, warna, tahun) dan behavior (maju, mundur, belok)
- Object merupakan instance dari sebuah class
- Pada sebuah class, state disebut juga sebagai field atau attribute, dan behavior dalam sebuah class berbentuk method

Object

Berikut ini adalah contoh bagaimana membuat sebuah object dari sebuah class:

```
MyClass myObject = new MyClass();
```

- **new MyClass()** disebut sebagai **instansiasi**, yaitu memanggil **constructor** dari class MyClass dengan keyword new. MyClass() adalah default constructor dari class MyClass
- **myObject** adalah nama dari object yang dibuat
- **MyClass** adalah tipe object, sesuai dengan nama class dari object

Example #1

```
class Mobil {  
    String merek = "Honda";  
    String warna = "Merah";  
    int tahun = 2020;  
  
    void maju() {  
        System.out.println("maju");  
    }  
  
    void mundur() {  
        System.out.println('mundur');  
    }  
}
```

Example #2

```
class MobilKu {  
    public static void main(String[] args) {  
        Mobil m = new Mobil();  
        System.out.println(m.merek);  
        System.out.println(m.warna + m.tahun);  
        m.maju();  
        m.mundur();  
    }  
}
```

Explanation #1

- Pada contoh class di atas, class Mobil merupakan Object Class, sedangkan class MobilKu merupakan Driver Class
- Driver Class adalah class yang memiliki main method
- Main method diperlukan untuk menjalankan program pada Java
- Pada main method biasanya terdapat instance dari object yang dibuat dari Object Class

Explanation #2

- Object Class merupakan class yang merepresentasikan object yang nantinya akan digunakan pada Driver Class
- Object Class dapat ditemukan pada Java API atau class yang dibuat tersendiri seperti class Mobil pada contoh di atas