

BAB II

OBJECT DAN CLASS

CLASS

Class adalah prototype, atau blueprint, atau rancangan yang mendefinisikan sebuah objek dan fungsi yang dapat dilakukan object tersebut (disebut method). Class berfungsi untuk menampung isi dari program yang akan di jalankan, di dalamnya berisi atribut / type data dan method untuk menjalankan suatu program.

Contoh Pembuatan Class :

Berikut adalah contoh dimana kita membuat sebuah Class Fruit

Fruit.java

```
public class Fruit {  
  
}
```

Kemudian buat class yang lain dengan nama class Main dimana akan kita gunakan untuk memanggil object Fruit

Main.java

```
public class Main {  
    Run | Debug  
    public static void main(String[] args) {  
  
    }  
}
```

Nah kita sudah berhasil membuat dua class yaitu class Main dan class Fruit, dalam penamaan Class sendiri lebih baik menggunakan *Upper Cammel Case* contohnya : *KucingAnggora*

OBJECT

Dalam paradigma OOP, object merupakan instance atau wujud nyata dari suatu class. Pemahaman mengenai object tidak kalah pentingnya pada konsep pemrograman berorientasi object. Salah satu kegunaan object adalah untuk mengakses berbagai fungsi dan metode pada class. Objek bisa dibuat lebih dari satu.

Contoh pemanggilan object di dalam class :

Fruit.java

```
public class Fruit {  
  
    public void cetakNama(String nama) {  
        System.out.println("Nama buah : " + nama);  
    }  
  
}
```

Di sini kita menambahkan method untuk mencetak nama buah

Main.java

```
public class Main {  
    Run | Debug  
    public static void main(String[] args) {  
  
        Fruit objekBuah = new Fruit();  
        objekBuah.cetakNama(nama: "Pisang");  
    }  
}
```

Kemudian membuat object untuk class Fruit di class Main dan disini kita bisa memanggil method milik class Fruit karena sudah dibuatkan Object-nya.

Kita juga dapat menambahkan beberapa method dan atribut lain di class Fruit, dalam hal ini atribut akan dibahas pada kelas selanjutnya.

```
public class Fruit {  
  
    String nama;  
    String warna;  
    int harga;  
  
    public void warnaBuah(String warnaHewan) {  
        warna = warnaHewan;  
    }  
  
    public void hargaBuah(int hargaBuah) {  
        harga = hargaBuah;  
    }  
  
    public void namaBuah(String namaBuah) {  
        nama = namaBuah;  
    }  
  
    public void detailBuah() {  
        System.out.println("Nama Buah : " + nama);  
        System.out.println("Warna : " + warna);  
        System.out.println("Harga Buah : " + harga);  
    }  
}
```

Pada gambar di atas kita membuat method tambahan yaitu *namaBuah*, *warnaBuah*, *hargaBuah*, dan mencetak hasilnya pada method *detailBuah*.

```
public class Main {  
    Run | Debug  
    public static void main(String[] args) {  
  
        Fruit objekBuah = new Fruit();  
        objekBuah.namaBuah(namaBuah: "Pisang");  
        objekBuah.warnaBuah(warnaHewan: "Kuning");  
        objekBuah.hargaBuah(hargaBuah: 5000);  
        objekBuah.detailBuah();  
    }  
}
```

Kemudian method-method pada objek buah dipanggil kembali di kelas Main.