	<p>Modul 4 Break, Continue, Return dan Pendahuluan Pemrograman Berorientasi Objek (OOP) Praktikum Pemrograman Berorientasi Objek Jurusan Teknik Informatika Fakultas Sains dan Teknologi</p> <hr/> <p>Dosen: Rifqi Syamsul Fuadi, ST.</p>
---	--

A. Tujuan

Dengan praktikum ini diharapkan mahasiswa dapat:

1. Mengetahui break, continue, dan return.
2. Menggunakan break, continue, dan return dalam suatu program.
3. Mengetahui objek, class, atribut, dan method.
4. Menggunakan objek, class, atribut, dan method dalam sebuah program.

B. Pokok Bahasan

1. Pengenalan Break, Continue, dan Return.
2. Penggunaan Break, Continue, dan Return.
3. Pengenalan awal pemrograman berorientasi objek.
4. Pengenalan Objek, Class, Atribut dan Method.

C. Dasar Teori

Pembahasan break, continue dan return mengarahkan dalam mempelajari *branching statement* dalam suatu program. *Branching statement* digunakan untuk mengatur jalannya pengeksekusian program. Dalam Java terdapat 3 *branching statement*, yakni break, continue, return.

1. Break

Break digunakan untuk menghentikan percabangan atau pengulangan. Jika terjadi break, pengulangan akan berhenti dan alur akan keluar dari loop. Dalam break dikenal ada 2 jenis break, yakni:

- Unlabeled break statement

```
public class break1{
    public static void main(String[] args){
        String names[]={"Beah", "Bianca", "Lance", "Belle","Nico", "Yza"};
        String searchName="Yza";
        boolean foundName=false;

        for(int i=0;i<names.length;i++){
            if(names[i].equals(searchName)){
                foundName=true;
                break;
            }
        }
        if(foundName){
            System.out.println(searchName+" ditemukan!");
        }
        else{
            System.out.println(searchName+" tidak ditemukan!");
        }
    }
}
```

- Labeled break statement

```
public class break2{
    public static void main(String[] args){
        int[][] numbers={{1,2,3},{4,5,6},{7,8,9}};
        int searchNum=5;
        boolean foundNum=false;

        searchLabel:
        for(int i=0;i<numbers.length;i++){
            for(int j=0;j<numbers[i].length;j++){
                if(searchNum==numbers[i][j]){
                    foundNum=true;
                    break searchLabel;
                }
            }
        }
        if(foundNum){
            System.out.println(searchNum+ " ada lho!");
        }
        else{
            System.out.println(searchNum+ " tidak ada lho!");
        }
    }
}
```

2. Continue

Jika dalam suatu program terdapat continue, statement akan dilewati dan alur akan kembali ke evaluasi kondisi. Dalam continue pun dikenal ada 2 jenis continue, yakni:

- Unlabeled continue statement

```
public class continue1{
    public static void main(String[] args){
        String names[]={"Beah", "Bianca", "Lance", "Beah"};
        int count=0;

        for(int i=0;i<names.length;i++){
            if(!names[i].equals("Beah")){
                continue;
            }
            count++;
        }

        System.out.println("There are " + count + " Beahs in the list");
    }
}
```

- Labeled continue statement

```
public class continue2{
    public static void main(String[] args){

        outerLoop:
        for( int i=0; i<5; i++ ){
            for( int j=0; j<5; j++ ){
                System.out.println("Inside for(j) loop"); //message1
                if( j == 2 ) continue outerLoop;
            }
            System.out.println("Inside for(i) loop"); //message2
        }
    }
}
```

3. Return

Return akan diperlukan untuk keluar dari method, dengan dijalankan perintah return, suatu method telah selesai dijalankan. Statement return memiliki dua bentuk, yakni: menggunakan sebuah nilai, dan tidak memberikan nilai.

PENGENALAN PEMROGRAMAN BERORIENTASI OBJEK

1. Objek

Objek adalah kesatuan entitas (benda), baik yang berwujud nyata ataupun hanya suatu sistem atau konsep yang memiliki sifat karakteristik dan fungsi. Semua benda yang ada di dunia nyata dapat dianggap sebagai sebuah objek. Dua karakteristik yang utama pada sebuah objek, yaitu:

- Setiap objek memiliki atribut sebagai status yang kemudian akan disebut sebagai state.
- Setiap objek memiliki tingkah laku yang kemudian akan disebut sebagai behaviour.

Contoh ilustrasi objek

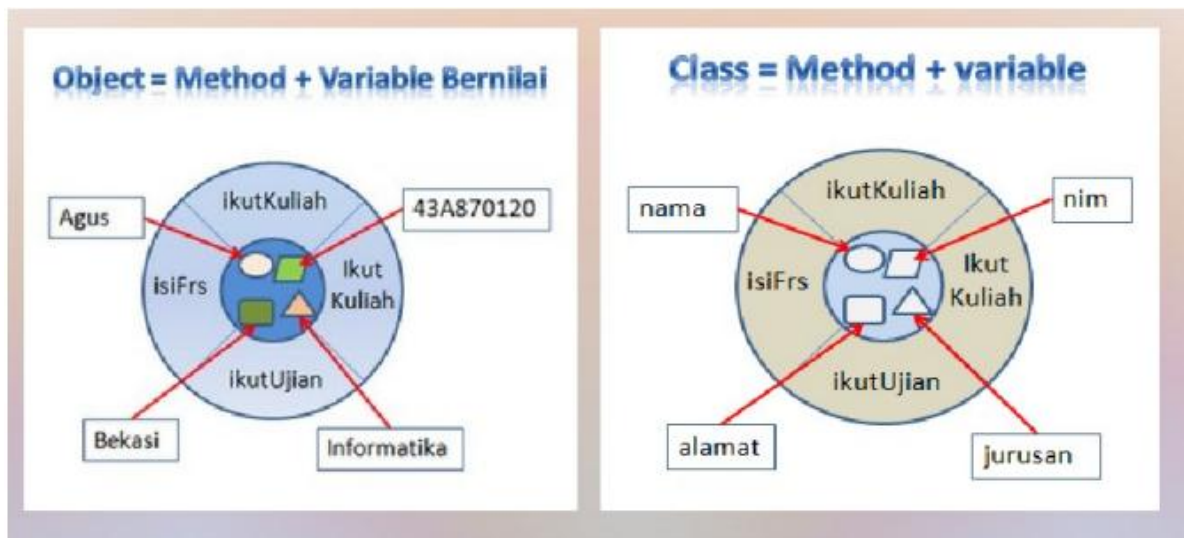


2. Kelas

Kelas adalah pemodelan dari objek yang berisi informasi (aturan) tentang sifat karakteristik (data) tingkah laku (method) yang dimiliki oleh objek tersebut. Kelas merupakan prototipe yang mendefinisikan variabel-variabel dan method-method secara umum. Sedangkan objek pada sisi yang lain merupakan instansiasi dari suatu kelas. Contoh ilustrasi dari sebuah kelas:



Berikut ini adalah ilustrasi untuk membedakan perbedaan antara class dan object. Supaya diketahui, pada real programming, istilah **attribut** merupakan **variable**, sedangkan **behaviour** berarti **method**. Pada gambar dikatakan bahwa objek merupakan **method** dan **variabel** yang bernilai, sedangkan **class** merupakan **method** dan **variabel** yang tidak bernilai.



3. Instance (Instans)

Instans adalah perwujudan. Dalam istilah pemrograman, objek merupakan instans (perwujudan) dari suatu kelas. Instansiasi adalah proses perwujudan kelas menjadi objek.

4. Data ↔ Variabel ↔ Property

Data adalah sifat karakteristik yang dimiliki objek. Dalam beberapa referensi, data dianalogikan juga sebagai variabel, property atau atribut.

5. Method ↔ Prosedur/Fungsi ↔ Behaviour

Method adalah prosedur atau fungsi yang dimiliki oleh suatu objek. Dalam beberapa referensi, method dianalogikan juga dengan istilah prosedur, fungsi dan behaviour.

D. Praktikum

Untuk lebih memudahkan dalam mempelajari kelas, objek, atribut dan method. Beberapa praktikum dibawah ini dapat dicoba sebagai perbandiannya.

Program 2.1 MobilBeraksi.java

```
public class Mobil {                                     Mobil.java
    String warna;
    int tahunProduksi;
}

public class MobilBeraksi{                               MobilBeraksi.java
    public static void main(String[] args){
        // Membuat object
        Mobil mobilku = new Mobil();

        /* memanggil atribut dan memberi nilai */
        mobilku.warna = "Hitam";
        mobilku.tahunProduksi = 2006;
        System.out.println("Warna: " + mobilku.warna);
        System.out.println("Tahun: " + mobilku.tahunProduksi);
    }
}
```

Program 2.2 Mobil2Beraksi.java

```
public class Mobil2{                                     Mobil2.java
    String warna;
    int tahunProduksi;
    void printMobil(){
        System.out.println("Warna: " + warna);
        System.out.println("Tahun: " + tahunProduksi);
    }
}

public class Mobil2Beraksi{                               Mobil2Beraksi.java
    public static void main(String[] args){
        // Membuat object
        Mobil2 mobilku = new Mobil2();

        /* memanggil atribut dan memberi nilai */
        mobilku.warna = "Hitam";
        mobilku.tahunProduksi = 2006;
        mobilku.printMobil();
    }
}
```

Program 2.3 MamaliaDarat.java

```
public class MamaliaDarat{

    public static void main(String[] args){

        // Mendeklarasikan dan menginisialisasi data
        String suaraHewan = "";

        System.out.println("\nDEMO KELAS, METHOD DAN DATA");
        System.out.println("-----\n");

        Kucing pussy = new Kucing();
        suaraHewan = pussy.mengeong();
        System.out.println("Pussy : " + suaraHewan);

        Anjing doggy = new Anjing();
        suaraHewan = doggy.menggonggong();
        System.out.println("Doggy : " + suaraHewan);

        Macan siKumbang = new Macan();
        suaraHewan = siKumbang.mengaum();
        System.out.println("Si Kumbang : " + suaraHewan);

    }
}

class Kucing {
    // Method yang mengembalikan suara kucing
    public String mengeong() {
        String suara = new String("Meoong...");
        return suara;
    }
}

class Anjing{
    // Method yang mengembalikan suara anjing
    public String menggonggong() {
        String suara = new String("Guk..Guk...");
        return suara;
    }
}

class Macan {
    // Method yang mengembalikan suara macan
    public String mengaum() {
        String suara = new String("Auuummm...");
        return suara;
    }
}
```

Program 2.4 DemoMatematika.java

```
class Matematika{  
    // Mendeklarasikan variabel kelas  
    private double hasil;  
  
    // Method penjumlahan  
    public double jumlah(double a, double b){  
        hasil = a + b;  
        return hasil;  
    }  
  
    // Method pengurangan  
    public double kurang(double a, double b){  
        hasil = a - b;  
        return hasil;  
    }  
  
    // Method perkalian  
    public double kali(double a, double b){  
        hasil = a * b;  
        return hasil;  
    }  
  
    // Method pembagian  
    public double bagi(double a, double b){  
        hasil = a / b;  
        return hasil;  
    }  
}  
  
public class DemoMatematika{  
  
    /**Main Method*/  
}
```

```

public static void main(String [] args){

    // Mendeklarasikan data
    double a = 100.0; double b = 3.0;
    double hasilJumlah, hasilKurang,
           hasilKali, hasilBagi;

    // Membuat objek "hitung" dari kelas Matematika
    Matematika hitung = new Matematika();

    hasilJumlah = hitung.jumlah( a, b );
    hasilKurang = hitung.kurang( a, b );
    hasilKali = hitung.kali( a, b );
    hasilBagi = hitung.bagi( a, b );

    // Mencetak hasil di console
    System.out.println("\nPROGRAM DEMO OPERASI MATEMATIKA");
    System.out.println("-----\n");

    System.out.println("Penjumlahan " + a + " + " + b +
                       " = " + hasilJumlah);
    System.out.println("\nPengurangan " + a + " - " + b +
                       " = " + hasilKurang);
    System.out.println("\nPerkalian " + a + " x " + b +
                       " = " + hasilKali);
    System.out.println("\nPembagian " + a + " / " + b +
                       " = " + hasilBagi);

}
}

```


TUGAS

- Membuat 2 kelas
- Membuat 5 atribut
- Membuat minimal 3 method, 2 method buatan dan 1 method main.
- Inputan berasal dari keyboard boleh menggunakan scanner ataupun parameter argumen.
- Lalu buat laporannya dalam format pdf dan kirim ke e-mail : rifqi@if.uinsgd.ac.id
- Subject : Tugas2_NIM_Kelas
- Paling telat hari Sabtu, 20 Feb 2016 jam 23.59 WIB.