

Fungsi 1

1. Kompetensi

1. Mahasiswa mampu memahami penggunaan fungsi *static* pada Java dengan parameter dan mengembalikan nilai.
2. Mahasiswa mampu membuat program menggunakan fungsi *static* dan mengeksekusi fungsi tersebut.

2. Ulasan Teori:

Pengertian Fungsi

Dalam pemrograman terdapat istilah fungsi, prosedur, dan method, yang ketiganya pada dasarnya adalah sama, di mana kita dapat menyebut ketiganya sebagai fungsi. **Prosedur** adalah sebutan untuk fungsi yang tidak mengembalikan nilai. Fungsi ini biasanya ditandai dengan kata kunci **void**. **Fungsi** adalah sebutan untuk fungsi yang mengembalikan nilai. **Method** adalah fungsi yang berada di dalam sebuah Class. Sebutan ini, biasanya digunakan pada **OOP**.

Ada 2 jenis fungsi di Java, yaitu fungsi *static* dan *non-static*. Fungsi *static* adalah fungsi yang dapat dieksekusi langsung tanpa harus melakukan instansiasi objek. Sedangkan fungsi *non-static* adalah fungsi yang harus dieksekusi dari objek yang sudah diinstansiasi, di mana ini berkaitan dengan OOP. Sehingga pada modul praktikum kali ini kita hanya membahas tentang **fungsi static** di Java. Cara mendeklarasikan fungsi *static* di Java adalah dengan menambahkan *keyword static*.

Sintaks Fungsi di Java

Fungsi harus dibuat atau ditulis di dalam *class*. Sintaks dasar penulisan fungsi adalah sebagai berikut:

```
static TypeDataKembalian namaFungsi() {  
    // statement  
}
```

Keyword static, artinya kita akan membuat fungsi static. **TypeDataKembalian** adalah tipe data dari nilai yang dikembalikan (*output*) setelah fungsi dieksekusi. Jika fungsi tersebut tidak mengembalikan *output*, maka TypeDataKembalian adalah **void**. Sedangkan, **namaFungsi()** adalah nama fungsi yang kita buat, ditulis dengan cara *camel case*. Contoh:

```
static void beriSalam() {  
    System.out.println("Halo! Selamat Pagi");  
}
```

Cara Menjalankan/Eksekusi Fungsi

Setelah kita membuat fungsi, selanjutnya kita bisa mengeksekusi fungsinya. Fungsi dapat dipanggil dari **fungsi *main*** atau dari **fungsi yang lainnya**. Contoh pemanggilan fungsi dalam fungsi *main*:

```
public static void main(String[] args){  
    beriSalam();  
}
```

Kode lengkapnya, silahkan dicoba pada bagian Praktikum di bawah.

Fungsi dengan Parameter

Parameter adalah variabel yang menampung nilai untuk diproses di dalam fungsi. Parameter berperan sebagai *input* untuk fungsi.

Struktur dasarnya seperti berikut ini:

```
static TipeDataKembalian namaFungsi(TipeData namaParameter,  
    TipeData namaParameterLain){  
    // statement  
}
```

Parameter ditulis di antara *parenthesis* (...) setelah nama fungsi. Bila terdapat lebih dari satu parameter, maka dipisah dengan tanda koma dan masing-masing parameter harus dideskripsikan tipe datanya.

Misalkan, kita membuat parameter bernama ucapan dengan tipe String. Sehingga kita bisa menggunakan variabel ucapan di dalam fungsi berikut ini:

```
static void beriUcapan(String ucapan){  
    System.out.println(ucapan);  
}
```

Cara eksekusi fungsinya adalah dengan memberikan nilai yang akan diinputkan sebagai parameter, bisa berupa variabel atau langsung nilainya. Contohnya:

```
String halo = "Hallo!";  
beriUcapan(halo);  
beriUcapan("Selamat datang di pemrograman Java");
```

Fungsi yang Mengembalikan Nilai

Pada kasus tertentu kita membutuhkan sebuah fungsi yang dapat mengembalikan nilai *output* sehingga bisa diolah pada proses berikutnya. Pengembalian nilai pada fungsi menggunakan keyword **return**.

```
static TipeDataKembalian namaFungsi(TipeData namaParameter) {  
    // statement  
    return variabelOutput;  
}
```

Contoh:

```
static int luasPersegi(int sisi){  
    int luas = sisi * sisi;  
    return luas;  
}
```

Pada contoh tersebut, kita membuat sebuah parameter bernama sisi. Kemudian fungsi akan mengembalikan output dengan tipe int (*integer*) dari variabel luas. Cara pemanggilan fungsi tersebut adalah seperti berikut:

```
System.out.println("Luas Persegi dengan sisi 5 = " + luasPersegi(5));  
int luasan = luasPersegi(6);
```

3. Langkah Praktikum

Praktikum 1

1. Buat *class* baru dengan nama **Greeting**.

```
1 class Greeting{  
2  
3 }
```

2. Buat fungsi **beriSalam** di dalam *class* tersebut.

```
1 class Greeting{  
2     static void beriSalam(){  
3         System.out.println("Halo! Selamat Pagi");  
4     }  
5 }
```



3. Buat fungsi **main** di dalam *class* tersebut, dan eksekusi fungsi **beriSalam** dari dalam fungsi *main*.

```

1 class Greeting{
2     static void beriSalam(){
3         System.out.println("Halo! Selamat Pagi");
4     }
5
6     public static void main(String[] args){
7         beriSalam();
8     }
9 }

```

Praktikum 2

1. Buat fungsi **beriUcapan** dengan sebuah parameter bertipe *String* di dalam *class* *Greeting*.

```

1 class Greeting{
2     static void beriSalam(){
3         System.out.println("Halo! Selamat Pagi");
4     }
5
6     static void beriUcapan(String ucapan){
7         System.out.println(ucapan);
8     }
9
10    public static void main(String[] args){
11        beriSalam();
12    }
13 }

```

2. Eksekusi fungsi **beriUcapan** dari dalam fungsi *main*.

```

1 class Greeting{
2     static void beriSalam(){
3         System.out.println("Halo! Selamat Pagi");
4     }
5
6     static void beriUcapan(String ucapan){
7         System.out.println(ucapan);
8     }
9
10    public static void main(String[] args){
11        beriSalam();
12        String salam = "Selamat datang di pemrograman Java";
13        beriUcapan(salam);
14    }
15 }

```

Praktikum 3

1. Buat *class* baru dengan nama **Persegi**.

```
1 class Persegi{  
2  
3 }
```

2. Buat fungsi **luasPersegi** di dalam *class* tersebut yang mengembalikan nilai luas (*int*) dan parameter masukan sisi (*int*).

```
1 class Persegi{  
2     static int luasPersegi(int sisi){  
3         int luas = sisi * sisi;  
4         return luas;  
5     }  
6 }
```

3. Buat fungsi **main** di dalam *class* tersebut, dan eksekusi fungsi **luasPersegi** dari dalam fungsi *main*.

```
1 class Persegi{  
2     static int luasPersegi(int sisi){  
3         int luas = sisi * sisi;  
4         return luas;  
5     }  
6  
7     public static void main(String[] args){  
8         int luasan = luasPersegi(5);  
9         System.out.println("Luas Persegi dengan sisi 5 = " + luasan);  
10    }  
11 }
```

Praktikum 4

1. Buat *class* baru dengan nama **Prak4**.
2. Buatlah fungsi **Kali** di dalam *class* tersebut yang mengembalikan nilai **H** (*int*) dan parameter masukan **C** dan **D** (*int*).

```
static int Kali(int C, int D) {  
    int H;  
    H = (C + 10) * (D + 19);  
    return H;  
}
```

3. Buatlah fungsi **Kurang** di dalam *class* tersebut yang mengembalikan nilai **X** (*int*) dan parameter masukan **A** dan **B** (*int*) dan memanggil fungsi **Kali**.



```
static int Kurang(int A, int B) {
    int X;
    A = A + 7;
    B = B + 4;
    X = Kali(A, B);
    return X;
}
```

4. Buat fungsi **main** di dalam *class* tersebut, dan eksekusi fungsi Kurang dari dalam fungsi *main*.

```
public static void main(String[] args) {
    int nilai1, nilai2;
    Scanner input = new Scanner (System.in);
    System.out.println("Masukkan Nilai 1:");
    nilai1=input.nextInt();
    System.out.println("Masukkan Nilai 2:");
    nilai2=input.nextInt();
    int hasil = Kurang(nilai1, nilai2);
    System.out.println("Hasil akhir adalah " + hasil);
}
```

Praktikum 5

1. Buat *class* baru dengan nama **Prak5**.
2. Buatlah fungsi Tampil (bertipe void) di dalam *class* tersebut dengan menggunakan dua jenis tipe data parameter yaitu string dan int

```
static void Tampil(String str, int... a) {
    System.out.println("String: " + str);
    System.out.println("Jumlah argumen/parameter: " + a.length);

    for (int i : a) {
        System.out.print(i + " ");
    }

    System.out.println();
}
```

3. Buat fungsi **main** di dalam *class* tersebut, dan eksekusi fungsi Tampil dari dalam fungsi *main*.

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

    Tampil("Daspro 2019", 100, 200);
    Tampil("Teknologi Informasi", 1, 2, 3, 4, 5);
    Tampil("Polinema");
}
```

Praktikum 6

1. Buat *class* baru dengan nama **Prak6**.
2. Buatlah program untuk menghitung luas persegi panjang dan volume balok tanpa menggunakan fungsi

```
public static void main(String[] args) {  
    Scanner input = new Scanner(System.in);  
  
    int p,l,t,L,vol;  
  
    System.out.println("Masukkan panjang");  
    p=input.nextInt();  
    System.out.println("Masukkan lebar");  
    l=input.nextInt();  
    System.out.println("Masukkan tinggi");  
    t=input.nextInt();  
  
    L=p*l;  
    System.out.println("Luas Persegi panjang adalah "+L);  
  
    vol=p*l*t;  
    System.out.println("Volume balok adalah "+vol);  
  
}
```

3. Program menghitung luas persegi dan volume balok diatas jika dibuatkan fungsi maka terdapat 3 fungsi yaitu hitungLuas, hitungVolume dan fungsi main, seperti dibawah ini:

Fungsi hitungLuas

```
static int hitungLuas (int pjg, int lb){  
    int Luas=pjg*lb;  
    return Luas;  
}
```

Fungsi hitungVolume

```
static int hitungVolume (int tinggi, int a, int b){  
    int volume= hitungLuas(a,b)*tinggi;  
    return volume;  
}
```

Fungsi main

```
public static void main(String[] args) {
    Scanner input = new Scanner (System.in);
    int p,l,t,L, vol;
    System.out.println("Masukkan panjang");
    p=input.nextInt();
    System.out.println("Masukkan lebar");
    l=input.nextInt();
    System.out.println("Masukkan tinggi");
    t=input.nextInt();

    L=hitungLuas(p,l);
    System.out.println("Luas Persegi Panjang adalah "+L);
    vol=hitungVolume(t,p,l);
    System.out.println("Volume Balok adalah "+vol);
}
```

4. Jelaskan jalannya program menghitung luas persegi dan volume balok diatas yang menggunakan fungsi!

Pertanyaan!

1. Berdasarkan praktikum 2 dan 3, jelaskan kapan suatu fungsi membutuhkan nilai kembalian (return)!
2. Pada praktikum 4 tambahkan satu fungsi yang digunakan untuk mengecek inputan nilai 1 dan nilai 2 harus minimal 0, kemudian panggil fungsi tersebut di fungsi main!
3. Jelaskan mengapa penulisan parameter di praktikum 5 di tulis dengan **int... a**!
4. Apakah output dari program dibawah ini kemudian jelaskan alur jalannya program tersebut!

```
1 public class programKu {
2     public static void TampilHinggaKei(int i) {
3         for (int j = 1; j <= i; j++) {
4             System.out.print(j);
5         }
6     }
7
8     public static int Jumlah (int bil1, int bil2) {
9         return (bil1 + bil2);
10    }
11
12    public static void TampilJumlah (int bil1, int bil2) {
13        TampilHinggaKei(Jumlah(bil1, bil2));
14    }
15
16    public static void main (String[] args) {
17        int temp = Jumlah(1, 1);
18        TampilJumlah(temp, 5);
19    }
20 }
```


Tugas

1. Buatlah sebuah *static method* yang bernama **Max3(int bil1, int bil2, int bil3)** yang menerima 3 buah parameter bilangan *integer* dan mengembalikan sebuah bilangan *integer* yang merupakan nilai maksimum diantara ketiga bilangan tersebut. Catatan: Anda boleh membuat *static method* lain selain Max3. Setelah itu, gunakanlah *static method* Max3 tersebut di *method* utama kalian (penggunaannya bebas).
2. Buatlah sebuah *class* Lingkaran yang di dalamnya terdapat fungsi untuk menghitung keliling lingkaran dan luas lingkaran.
3. Terdapat array A satu dimensi dengan ilustrasi seperti gambar dibawah ini yang dibuat didalam fungsi main (belum ada isinya)

0	1	2	3	4	5	6	7	8	9	10

Buatlah sebuah fungsi untuk mengisi array tersebut, sehingga isinya menjadi seperti berikut:

0	1	2	3	4	5	6	7	8	9	10
1		2		3		4		5		6

4. Buatlah program untuk mengisi array B sampai penuh dengan tipe data berupa int (nilai ujian 10 mahasiswa), dimana proses penginputan dan pengisiannya kedalam array dilakukan dalam sebuah fungsi. Selanjutnya buatlah lainnya yaitu untuk menghitung nilai rata-rata array tersebut (nilai rata-rata ujian mahasiswa). Dan cetak nilai rata-rata tersebut dimana instruksi mencetaknya berada di fungsi main.
5. Berdasarkan soal no 4, tambahkan sebuah fungsi lain untuk mencari nilai terbesar dan terkecil dari isi array tersebut. Dan cetak nilai terbesar dan terkecil tersebut, dimana instruksi mencetaknya berada pada fungsi main.