

PERTEMUAN 11

FUCNTION



19_Muhammad Alif Febriansyah

2341720025

Teknik Informatika

Teknologi Informasi

Politeknik Negeri Malang

2023/2024

Percobaan 1

Kode program

```
1 public class TerimaKasih19 {
2
3     public static void UcapanTerimaKasih() {
4         System.out.println("Thank you for being the best teacher in the world.\n"+"You inspired in me a love for learning and made me feel like i could ask you anything.");
5     }
6     public static void main(String[] args) {
7         UcapanTerimaKasih();
8     }
9 }
```

Output

```
PS D:\Polinema\semester 1\Daspro> java TerimaKasih19
Thank you for being the best teacher in the world.You inspired in me a love for learning and made me
feel like i could ask you anything.
```

Pertanyaan

1. Apakah fungsi tanpa parameter selalu digunakan untuk fungsi “void” saja?
Tidak. Fungsi tanpa parameter bisa digunakan untuk melakukan tugas tertentu tanpa memerlukan informasi tambahan dari luar. Meskipun fungsi tanpa parameter sering digunakan untuk fungsi yang tidak mengembalikan nilai (void).
2. Apakah bisa kalimat “Thank you for.....dst” dituliskan tanpa fungsi UcapanTerimaKasih? modifikasi kode program tersebut untuk dapat menampilkan suatu kalimat tanpa dan dengan menggunakan fungsi!
Ya, berikut adalah modifikasi kodenya

```
public class TerimaKasih19 {

    public static void main(String[] args) {
        System.out.println("Thank you for being the best teacher in the world.\n"
            + "You inspired in me a love for learning and made me feel like I
could ask you anything.");
    }
}
```
3. Apakah keuntungan menggunakan fungsi di dalam program?
Agar kode lebih efektif dan terstruktur

Percobaan 2

Kode program

```

1 public class TerimaKasih19 {
2
3     public static void UcapanTerimaKasih() {
4         System.out.println("Thank you for being the best teacher in the world.\n"+"You inspired in me a love for learning and made me feel like i could ask you anything.");
5     }
6     //public static void main(String[] args) {
7         //UcapanTerimaKasih();
8         //}
9     public static void UcapanTambahkan(String ucapan) {
10         System.out.println(ucapan);
11     }
12     public static void main(String[] args) {
13         UcapanTerimaKasih();
14         String ucapan = "Terima Kasih Pak... Bu... Semoga Sehat Selalu";
15         UcapanTambahkan(ucapan);
16     }
17 }

```

Output

```

TerimaKasih19
Thank you for being the b
est teacher in the world.
You inspired in me a love for learning and made me feel like i could ask you anything.
Terima Kasih Pak... Bu... Semoga Sehat Selalu
PS D:\Polinema\semester 1\Daspro>

```

Pertanyaan

1. Apakah kegunaan suatu parameter di dalam fungsi?
untuk memanggil nilai yang diterima oleh fungsi
2. Apakah parameter sama dengan variabel? jelaskan!
Tidak, jelas variable dan parameter berbeda. Variable digunakan di berbagai konteks dalam pemrograman untuk menyimpan nilai, parameter spesifik untu fungsi dan digunakan sebagai sarana untuk mengirim data ke dalam fungsi saat dipanggil.
3. Apakah parameter di dalam Bahasa pemrograman Java hanya digunakan untuk melewati data input saja? Bagaimana dengan data output?
Tidak, parameter dapat digunakan untuk mengembalikan data output melalui tipe data pengembalian.

Percobaan 3

Kode program

```

1  import java.util.Scanner;
2  public class Ucapan19 {
3
4      public static String PenerimaUcapan() {
5          Scanner sc = new Scanner(System.in);
6          System.out.println("Tuliskan NAMA orang yang ingin Anda beri ucapan: ");
7          String namaOrang = sc.nextLine();
8          sc.close();
9          return namaOrang;
10     }
11     public static void main(String[] args) {
12         String nama = PenerimaUcapan();
13         System.out.println("Thank you "+nama+"\nMay the force be with you. ");
14     }
15 }

```

Output

```

Tuliskan NAMA orang yang ingin Anda beri ucapan: Bu Triana
Thank you Bu Triana
May the force be with you.
PS D:\Polinema\semester 1\Daspro>

```

Pertanyaan

1. Jelaskan mengenai, kapan suatu fungsi membutuhkan nilai kembalian (return)!
Return digunakan saat fungsi tidak memiliki void atau sebaliknya
2. Apakah bisa ditambahkan statement System.out.println di dalam fungsi dengan nilai kembalian? Apa pengaruh nya?
Bisa, kita dapat melihat hasil di berbagai bagian program
3. Apakah bisa fungsi tanpa nilai kembalian di dalam fungsi main dipanggil tanpa dilewatkan ke variabel? seperti pada percobaan 1? Jelaskan!
Bisa, fungsi tanpa nilai kembalian dapat dipanggil didalam fungsi 'main' tanpa harus menyimpan hasilnya ke dalam variable. Fungsi tanpa nilai kembalian biasanya digunakan untuk tugas tugas seperti mencetak output, memodifikasi variable diluar fungsi atau menjalankan aksi tertentu tanpa memerlukan hasil yang dikembalikan.

Percobaan 4

Kode program

```

1 import java.util.Scanner;
2 public class UcapanTerimaKasih19 {
3
4     public static String PenerimaUcapan() {
5         Scanner sc = new Scanner(System.in);
6         System.out.println("Tuliskan NAMA orang yang ingin Anda beri ucapan: ");
7         String namaOrang = sc.nextLine();
8         sc.close();
9         return namaOrang;
10    }
11    public static void UcapanTerimaKasih() {
12        String nama = PenerimaUcapan();
13        System.out.println("Thank you "+nama+" for beong the best teacher in the world.\n"+"You inspired me in a live for learning and made me feel like i could ask you anything.");
14    }
15    public static void main(String[] args) {
16        UcapanTerimaKasih();
17    }
18 }

```

Output

```

Tuliskan NAMA orang yang ingin Anda beri ucapan:
Bu Triana
Thank you Bu Triana for beong the best teacher in the world.
You inspired me in a live for learning and made me feel like i could ask you anything.
PS D:\Polinema\semester 1\Daspro>

```

Pertanyaan

1. Berdasarkan uji coba pada percobaan 4, manakah fungsi yang akan eksekusi pertama kali? Jelaskan!
 1. Program dimulai dengan eksekusi fungsi `main`.
 2. Di dalam fungsi `main`, terdapat pemanggilan `UcapanTerimaKasih()`.
 3. Selanjutnya, fungsi `UcapanTerimaKasih` dieksekusi.
 4. Di dalam fungsi `UcapanTerimaKasih`, terdapat pemanggilan `PenerimaUcapan()`.
 5. Fungsi `PenerimaUcapan` dieksekusi, yang meminta input dari pengguna untuk menuliskan nama orang yang ingin diberi ucapan.
 6. Setelah itu, hasil input tersebut digunakan untuk mencetak ucapan terima kasih di dalam fungsi `UcapanTerimaKasih`.
2. Manakah tata cara penulisan fungsi benar di dalam class? Di atas fungsi main ataukah dibawah fungsi main? Jelaskan!

Dalam java, penulisan fungsi dapat dilakukan di atas ataupun di bawah main selama semuanya berada dalam satu class
3. Modifikasi program di atas dengan menambahkan fungsi UcapanTambahan() dengan input parameter String. Fungsi UcapanTambahan() berisi ucapan tambahan yang ingin disampaikan kepada penerima ucapan.

Berikut adalah modifikasi program untuk menambahkan fungsi ucapantambahan

```
import java.util.Scanner;
```

```

public class UcapanTerimaKasih19 {

    public static String PenerimaUcapan() {
        Scanner sc = new Scanner(System.in);
        System.out.println("Tuliskan NAMA orang yang ingin Anda beri ucapan: ");
        String namaOrang = sc.nextLine();
        sc.close();
        return namaOrang;
    }
}

```

```

    public static void UcapanTerimaKasih() {
        String nama = PenerimaUcapan();
        String ucapanTambahan = UcapanTambahan("Your additional message goes
here.");
        System.out.println("Thank you " + nama + " for being the best teacher in the
world.\n"
            + "You inspired me in a love for learning and made me feel like I
could ask you anything.\n"
            + ucapanTambahan);
    }

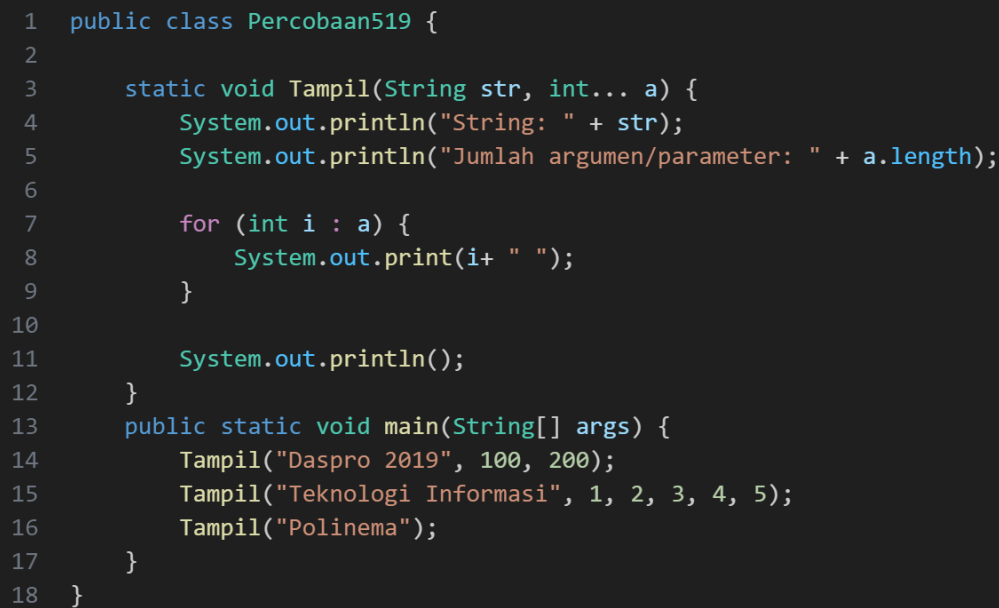
    public static String UcapanTambahan(String tambahan) {
        return tambahan;
    }

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

```

Percobaan 5

Kode program



```

1  public class Percobaan519 {
2
3      static void Tampil(String str, int... a) {
4          System.out.println("String: " + str);
5          System.out.println("Jumlah argumen/parameter: " + a.length);
6
7          for (int i : a) {
8              System.out.print(i+ " ");
9          }
10
11         System.out.println();
12     }
13     public static void main(String[] args) {
14         Tampil("Daspro 2019", 100, 200);
15         Tampil("Teknologi Informasi", 1, 2, 3, 4, 5);
16         Tampil("Polinema");
17     }
18 }

```

Output

```
String: Daspro 2019
Jumlah argumen/parameter:
 2
100 200
String: Teknologi Informa
si
Jumlah argumen/parameter:
 5
1 2 3 4 5
String: Polinema
Jumlah argumen/parameter: 0
PS D:\Polinema\semester 1\Daspro>
```

Pertanyaan

1. Jelaskan mengapa penulisan parameter di praktikum 5 di tulis dengan int... a!
Untuk mengirim sejumlah argument dari jenis yang sama ke sebuah metode dan java akan mengemasnya menjadi sebuah array.
2. Sebutkan kegunaan varargs dalam implementasi kode program untuk menyelesaikan permasalahan dalam dunia nyata! (minimal 3)
 - a. Operasi matematika dengan jumlah parameter yang variative
 - b. Pemrosesan data untuk statistic atau analisis
 - c. Membuat fungsi untuk menyimpan data
3. Bisakah kita menggunakan dua tipe data varargs dalam satu fungsi? Berikan contohnya!

Tidak, contohnya

```
```java
```

```
public class Contoh {
```

```
 // Ini akan menyebabkan kesalahan kompilasi
```

```
 public static void fungsiDuaVarargs(String... kata, int... angka) {
```

```
 // Implementasi
```

```
 }
```

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

```
 // Panggilan fungsi yang tidak valid
```

```
 fungsiDuaVarargs("A", "B", "C", 1, 2, 3);
```

```
 }
```

```
}
```

```
```
```

Pada contoh di atas, fungsi `fungsiDuaVarargs` mencoba menggunakan dua varargs, satu untuk string (`String... kata`) dan satu untuk integer (`int... angka`). Namun, ini tidak diperbolehkan dalam Java, dan program ini akan menghasilkan kesalahan kompilasi.

Kesalahan yang mungkin Anda temui adalah:

```
```
```

```
Error: illegal start of expression
```

```
```
```

Hal ini terjadi karena definisi varargs di Java hanya mengizinkan satu varargs dalam satu parameter fungsi.

Percobaan 6

Kode program

```
1  import java.util.Scanner;
2  public class Percobaan619 {
3
4      public static void main(String[] args) {
5          Scanner input = new Scanner(System.in);
6          int p,l,t,L,vol;
7          System.out.println("Masukkan panjang");
8          p=input.nextInt();
9          System.out.println("Masukkan lebar");
10         l=input.nextInt();
11         System.out.println("Masukkan tinggi");
12         t=input.nextInt();
13
14         L=hitungLuas (p,l);
15         System.out.println("Luas persegi panjang adalah "+L);
16         vol=hitungVolume(t,p,l);
17         System.out.println("Volume Balok adalah "+vol);
18
19     }
20     static int hitungLuas (int pjg, int lb) {
21         int Luas=pjg*lb;
22         return Luas;
23     }
24     static int hitungVolume (int tinggi, int a, int b) {
25         int volume= hitungLuas(a,b)*tinggi;
26         return volume;
27     }
28 }
```

output

```
Percobaan619
Masukkan panjang
5
Masukkan lebar
5
Masukkan tinggi
5
Luas persegi panjang adalah 25
Volume Balok adalah 125
PS D:\Polinema\semester 1\Daspro>
```


Pertanyaan

1. Sebutkan tahapan dan urutan eksekusi pada percobaan 6 di atas!
Berikut adalah alur kode program yang diberikan:
 - a. Program dimulai dengan mendeklarasikan class bernama 'Percobaan619'.
 - b. Di dalam class tersebut, ada fungsi 'main' yang merupakan titik awal eksekusi program. Variabel-variabel 'p', 'l', 't', 'L', dan 'vol' dideklarasikan sebagai variabel untuk menyimpan nilai panjang, lebar, tinggi, luas, dan volume.
 - c. Program kemudian membuat objek 'Scanner' bernama 'input' untuk menerima input dari pengguna.
 - d. Pengguna diminta untuk memasukkan nilai panjang ('p'), lebar ('l'), dan tinggi ('t') menggunakan perintah 'input.nextInt()'.
 - e. Fungsi 'hitungLuas' dipanggil dengan parameter panjang ('p') dan lebar ('l'). Fungsi ini menghitung luas persegi panjang dan mengembalikan nilai luas ('Luas').
 - f. Luas yang dihitung dari fungsi 'hitungLuas' kemudian dicetak menggunakan perintah 'System.out.println'.
 - g. Fungsi 'hitungVolume' dipanggil dengan parameter tinggi ('t'), panjang ('p'), dan lebar ('l'). Fungsi ini menggunakan fungsi 'hitungLuas' untuk menghitung luas, kemudian mengalikan luas dengan tinggi untuk menghitung volume dan mengembalikan nilai volume ('volume').
 - h. Volume yang dihitung dari fungsi 'hitungVolume' kemudian dicetak menggunakan perintah 'System.out.println'.
 - i. Program selesai dieksekusi.
2. Apakah output dari program dibawah ini kemudian jelaskan alur jalannya program tersebut!
Output dari program di atas adalah: 1
Alur jalannya program tersebut adalah sebagai berikut:
 1. Program dimulai dengan mendefinisikan kelas "programky".
 2. Di dalam kelas tersebut, terdapat method "o" yang memiliki dua parameter bertipe int, yaitu "bic" dan "bila".
 3. Di dalam method "o", terdapat pernyataan "return (bic + bila)" yang mengembalikan hasil penjumlahan dari kedua parameter tersebut.
 4. Kemudian, program mendefinisikan method "Tarpilminckaked" yang memiliki satu parameter bertipe int, yaitu "Jumlah".
 5. Di dalam method "Tarpilminckaked", terdapat pernyataan "System.out.print(Jumlah)" yang mencetak nilai dari parameter "Jumlah".
 6. Selanjutnya, program mendefinisikan method "op" yang memiliki satu parameter bertipe String, yaitu "l".
 7. Di dalam method "op", tidak ada pernyataan atau kode program yang dijalankan.
 8. Terakhir, program mencetak angka 1 menggunakan pernyataan "System.out.print(j)" di luar dari semua method yang telah didefinisikan sebelumnya. Sehingga, output dari program tersebut adalah angka 1.
3. Pada saat apakah fungsi yang kita buat harus menggunakan parameter atau tidak? Pada saat apakah fungsi yang kita buat harus memiliki nilai kembalian atau tidak? Jelaskan!

Pemilihan untuk menggunakan parameter atau nilai kembalian dalam fungsi tergantung pada kebutuhan spesifik dari tugas yang akan dijalankan oleh fungsi tersebut. Berikut adalah beberapa panduan umum:

Penggunaan Parameter:

1. Mengirim Data ke Fungsi:

- Jika fungsi memerlukan data sebagai input untuk melakukan tugas tertentu, parameter digunakan untuk menerima nilai tersebut.

2. ****Fleksibilitas dan Reusabilitas:****

- Penggunaan parameter membuat fungsi lebih fleksibel karena dapat digunakan dengan nilai yang berbeda pada setiap pemanggilan.

- Meningkatkan reusabilitas karena fungsi dapat diaplikasikan pada berbagai situasi dengan nilai input yang berbeda.

Penggunaan Nilai Kembalian:

1. Memberikan Output:

- Fungsi yang memberikan hasil atau output dari suatu operasi umumnya memiliki nilai kembalian.

2. Memproses Data dan Memberikan Informasi:

- Fungsi yang memproses data dan memberikan informasi juga sering menggunakan nilai kembalian.

3. Keterangan Kesalahan (Exception Handling):

- Nilai kembalian dapat digunakan untuk memberikan informasi tentang keberhasilan atau kegagalan suatu operasi.