

**LAPORAN PRAKTIKUM
DASAR PEMROGRAMAN**

**PERTEMUAN 13
FUNGSI 1**

HIKMAH ALDRIN ABDILLAH

NIM: 2341720049



**PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
NOVEMBER 2023**

2.1 Percobaan 1: Membuat Fungsi Tanpa Parameter

Pertanyaan!

1. Apakah fungsi tanpa parameter selalu digunakan untuk fungsi “void” saja?

Jawab:

Tidak selalu. Fungsi tanpa parameter dapat digunakan dalam berbagai konteks, tidak hanya untuk fungsi yang mengembalikan 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!

Jawab:

```
public class Terimakasih_14 {  
    // Dengan fungsi  
    public static void UcapanTerimakasih() {  
        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.");  
    }  
  
    public static void main(String[] args) {  
        // tidak menggunakan fungsi  
        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.");  
        // pemanggilan fungsi  
        // UcapanTerimakasih();  
    }  
}
```

3. Apakah keuntungan menggunakan fungsi di dalam program?

Jawab :

Lebih memudahkan dalam pengelolaan code, kode jadi mudah terbaca, lebih simple dan tidak ribet, Kode lebih terstruktur.

2.2 Percobaan 2: Membuat Fungsi Dengan Parameter

Pertanyaan!

1. Apakah kegunaan suatu parameter di dalam fungsi?

Jawab:

-Memungkinkan nilai yang diberikan kepada fungsi ketika fungsi tersebut dipanggil

2. Apakah parameter sama dengan variabel? jelaskan!

Jawab:

- sama sama bisa menyimpan nilai, perbedaannya adalah Parameter berperan sebagai input untuk fungsi. Parameter hanya berlaku dan dapat diakses di dalam fungsi atau metode tempat mereka didefinisikan. Variabel dapat digunakan di berbagai bagian program, Variabel juga bisa dideklarasikan di berbagai tempat dalam program dan digunakan untuk menyimpan nilai yang dapat diakses dan dimanipulasi.

3. Apakah parameter di dalam Bahasa pemrograman Java hanya digunakan untuk melewati data input saja? Bagaimana dengan data output?

Jawab :

Parameter dalam java tidak hanya digunakan untuk melewati data input, tetapi juga dapat digunakan untuk mengembalikan data output dari suatu metode (fungsi). Dalam Java, Anda dapat mengembalikan nilai dari suatu metode menggunakan tipe data pengembalian (return type) dan kata kunci "return".

2.3 Percobaan 3: Membuat Fungsi dengan Nilai Kembalian

Pertanyaan!

1. Jelaskan mengenai, kapan suatu fungsi membutuhkan nilai kembalian (return)!

Jawab:

Suatu fungsi membutuhkan nilai kembalian (return) ketika hasil dari operasi yang dilakukan oleh fungsi tersebut perlu dikembalikan kepada pemanggil fungsi

2. Apakah bisa ditambahkan statement `System.out.println` di dalam fungsi dengan nilai kembalian? Apa pengaruh nya?

Jawab:

Iya, Kita bisa menambahkan statement **`System.out.println`** di dalam fungsi yang

mengembalikan nilai. Namun, statement **System.out.println** akan memunculkan hasil di konsol atau output, dan tidak akan mempengaruhi nilai kembalian dari fungsi itu sendiri.

3. Apakah bisa fungsi tanpa nilai kembalian di dalam fungsi main dipanggil tanpa dilewatkan ke variabel? seperti pada percobaan 1? Jelaskan!

Jawab:

fungsi tanpa nilai kembalian (void functions) dapat dipanggil di dalam fungsi **main** tanpa perlu menyimpan hasil panggilan ke dalam variabel. Fungsi seperti itu biasanya digunakan untuk melakukan tugas atau operasi tanpa mengembalikan nilai yang harus disimpan.

2.4 Percobaan 4: Fungsi yang dapat meng-Calling Fungsi Lain

Pertanyaan!

1. Berdasarkan uji coba pada percobaan 4, manakah fungsi yang akan eksekusi pertama kali? Jelaskan!

Jawab:

Fungsi yang akan dieksekusi pertama kali adalah fungsi main. Karena dalam java eksekusi program akan dimulai dari main. Disaat program di run maka java akan memulai me – run program dari fungsi main, pada percobaan 4 fungsi main dimulai dari metode `UcapanTerimaKasih();`. Dan setelah fungsi main di eksekusi program akan lanjut ke static void `UcapanTerimaKasih();`.

2. Manakah tata cara penulisan fungsi benar di dalam class? Di atas fungsi main ataukah dibawah fungsi main? Jelaskan!

Jawab:

Jadi tata cara penulisan fungsi (metode) sebenarnya tidak ada aturan yang ketat apakah harus di atas atau di bawah fungsi main. Namun, secara umum, sebuah program Java akan lebih mudah dibaca dan dimengerti jika fungsi-fungsi yang lebih kecil atau pendukung didefinisikan di atas fungsi main. Ini membantu membentuk struktur yang bersih dan membuat kode lebih terorganisir.

3. Modifikasi program di atas dengan menambahkan fungsi `UcapanTambahan()` dengan input parameter String. Fungsi `UcapanTambahan()` berisi ucapan tambahan yang ingin disampaikan kepada penerima ucapan.

Jawab:

```
import java.util.Scanner;

public class UcapanTerimakasih_14 {
    public static String PenerimaUcapan() {
        Scanner Input14 = new Scanner(System.in);
        System.out.println("Tuliskan NAMA orang yang ingin Anda berikan ucapan: ");
        String namaOrang = Input14.nextLine();
        Input14.close();
        return namaOrang;
    }

    public static void UcapanTerimakasih() {
        String nama = PenerimaUcapan();
        System.out.println("Thank you " + nama + " 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.");
    }

    public static void UcapanTambahkan(String ucapan) {
        System.out.println("Ucapan Tambahan : " + ucapan);
    }

    public static void main(String[] args) {
        UcapanTerimakasih();
        UcapanTambahkan("You've made a great impact on my life ! Thank you sir");
    }
}
```

2.5 Percobaan 5: Fungsi Varargs

Pertanyaan!

1. Jelaskan mengapa penulisan parameter di praktikum 5 di tulis dengan int... a!

Jawab:

Varargs memungkinkan suatu metode menerima jumlah argumen yang bervariasi, maka dari itu parameter a bisa menerima argument 0 dan lebih dari itu.

2. Sebutkan kegunaan varargs dalam implementasi kode program untuk menyelesaikan permasalahan dalam dunia nyata! (minimal 3)

jawab:

- Pembuatan fungsi kalkulator
- Pembuatan sistem log
- Ketika membuat objek untuk representasi data, seperti titik dalam koordinat, yang bisa memiliki jumlah koordinat yang berbeda.

3. Bisakah kita menggunakan dua tipe data varargs dalam satu fungsi? Berikan contohnya!

Jawab:

```
static void Tampil(String str, int ... a) {
```

Bisa, menggunakan 2 tipe varargs dalam satu fungsi seperti diatas.

2.6 Percobaan 6: Pembuatan Kode Program, Dengan Fungsi versus Tanpa Fungsi

Pertanyaan!

1. Sebutkan tahapan dan urutan eksekusi pada percobaan 6 di atas!

Jawab:

- Urutan eksekusi program:
- Import package java.util.Scanner untuk mengambil input dari pengguna.
- Deklarasi Fungsi hitung Luas
- Deklarasi Fungsi hitung Volume
- Fungsi main
- Input dari Pengguna
- Perhitungan Luas dan Volume
- Output

2. Apakah output dari program dibawah ini kemudian jelaskan alur jalannya program tersebut!

Jawab:

- Pada metode main, temp diinisialisasi dengan hasil penjumlahan dari 1 dan 2, yaitu 3.
- Kemudian, metode TampilJumlah dipanggil dengan parameter temp 3 dan 5.
- Di dalam metode TampilJumlah, Jumlah(bil1, bil2) dipanggil, yang mengembalikan hasil penjumlahan 3 dan 5, yaitu 8.
- Selanjutnya, metode TampilHinggaKei dipanggil dengan hasil penjumlahan sebagai argumen. Maka, loop for dari 1 hingga 8 dieksekusi dan mencetak nilai j.

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!

Jawab:

Fungsi harus menggunakan parameter disaat membutuhkan input dari luar untuk diproses dalam fungsi. Namun, fungsi tidak perlu mempunyai nilai atau nilai balik dari suatu proses. Fungsi harus memiliki nilai kembalian jika memerlukan nilai balik atau output dari suatu proses.

3. Tugas

1. Buatlah sebuah class KubusNoAbsen yang di dalamnya terdapat fungsi untuk menghitung volume kubus dan luar permukaan kubus!

```
import java.util.Scanner;

public class Kubus14 {
    public static int hitungVolume(int sisi) {
        return sisi * sisi * sisi;
    }

    // Fungsi untuk menghitung luas permukaan kubus
    public static int hitungLuasPermukaan(int sisi) {
        return 6 * sisi * sisi;
    }

    // Fungsi Main
    public static void main(String[] args) {
        Scanner input14 = new Scanner(System.in);
        int s, LP, Vol;

        System.out.print("Masukkan panjang sisi :");
        s = input14.nextInt();

        LP = hitungLuasPermukaan(s);
        System.out.println("Luas Persegi adalah : " + LP);
    }
}
```

```

    Vol = hitungVolume(s);
    System.out.println("Volume Kubus adalah :" + Vol);

}
}

```

2. Buatlah program untuk mengisi array 2 dimensi nilai tugas lima mahasiswa sesuai dengan studi kasus pada tugas pada materi teori sebagai berikut :

	Minggu ke 1	Minggu ke 2	Minggu ke 3	Minggu ke 4	Minggu ke 5	Minggu ke 6	Minggu ke 7
Sari	20	19	25	20	10	0	10
Rina	30	30	40	10	15	20	25
Yani	5	0	20	25	10	5	45
Dwi	50	0	7	8	0	30	60
Lusi	15	10	16	15	10	10	5

Tambahkan fungsi untuk mengambil informasi dari data diatas dengan rincian sebagai berikut :

- Fungsi untuk meninputkan data nilai mahasiswa
- Fungsi untuk menampilkan seluruh nilai mahasiswa mulai dari minggu pertama sampai ketujuh
- Fungsi untuk mencari pada hari keberapakah terdapat nilai tertinggi dibanding hari lain dari keseluruhan mahasiswa
- Fungsi untuk menampilkan mahasiswa yang memiliki nilai tertinggi (tampilkan pula keterangan nilai dari minggu ke-)

3. Modifikasi program tugas no 2 dengan memastikan terdapat input dari user untuk menentukan jumlah mahasiswa dan juga jumlah tugas!

```

import java.util.Scanner;

public class NilaiMahasiswa_14 {
    static String[] namaMhs;
    static int[][] nilaiTugas = new int[5][7]; // array 2
    dimensinya

    public static void inputNilai() {
        Scanner input14 = new Scanner(System.in);

        System.out.print("Masukkan jumlah mahasiswa: ");
        int jumlahMhs = input14.nextInt();
        namaMhs = new String[jumlahMhs];
    }
}

```



```

        for (int i = 0; i < namaMhs.length; i++) {
            System.out.println("Masukkan Nama mahasiswa :");
            namaMhs[i] = input14.next();
            System.out.println("Masukkan Nilai Tugas Mahasiswa " +
namaMhs[i] + ": ");
            for (int j = 0; j < nilaiTugas[i].length; j++) {
                nilaiTugas[i][j] = input14.nextInt();
            }
        }
        System.out.println();
    }

    // Fungsi untuk menampilkan seluruh nilai mahasiswa mulai dari
    minggu pertama
    // hingga minggu ketujuh
    public static void tampilkanNilaiMahasiswa() {
        System.out.println("\nSeluruh Nilai Mahasiswa :");
        System.out.printf("| %-10s | %-10s | %-10s | %-10s | %-10s |
%-10s | %-10s | %-10s |\n", "NAMA", "MINGGU 1",
        "MINGGU 2", "MINGGU 3", "MINGGU 4", "MINGGU 5", "MINGGU
6", "MINGGU 7");

        for (int i = 0; i < namaMhs.length; i++) {
            System.out.printf("| %-10s", namaMhs[i]);

            for (int j = 0; j < nilaiTugas[i].length; j++) {
                System.out.printf(" | %-10s", nilaiTugas[i][j]);
            }

            System.out.println(" |");
        }

        System.out.println();
    }

    // Fungsi untuk mencari pada hari keberapakah terdapat nilai
    tertinggi dari
    public static int mingguNilaiTertinggi() {
        int mingguTertinggi = 0;
        int nilaiTertinggi = 0;

        for (int j = 0; j < nilaiTugas[0].length; j++) {
            int totalNilaiPerminggu = 0;
            for (int i = 0; i < namaMhs.length; i++) {
                totalNilaiPerminggu += nilaiTugas[i][j];
            }
        }
    }

```

```

    }
    if (totalNilaiPerminggu > nilaiTertinggi) {
        nilaiTertinggi = totalNilaiPerminggu;
        mingguTertinggi = j + 1;
    }
}
return mingguTertinggi;
}

// Fungsi untuk menampilkan mahasiswa yang memiliki nilai
// tertinggi
public static void mhsNilaiTertinggi(int minggu) {
    int nilaiTertinggi = 0;
    String mhsNilaiTertinggi = "";

    for (int i = 0; i < namaMhs.length; i++) {
        if (nilaiTugas[i][minggu - 1] > nilaiTertinggi) {
            nilaiTertinggi = nilaiTugas[i][minggu - 1];
            mhsNilaiTertinggi = namaMhs[i];
        }
    }

    System.out.println("Mahasiswa dengan nilai tertinggi adalah "
+ mhsNilaiTertinggi + " dengan nilai "
+ nilaiTertinggi + ", pada minggu ke-" + minggu + ".");
}

public static void main(String[] args) {
    inputNilai();
    tampilkanNilaiMahasiswa();
    int mingguTertinggi = mingguNilaiTertinggi();
    System.out.println("Pada minggu ke-" + mingguTertinggi
+ " terdapat nilai mahasiswa yang lebih tinggi
dibandingkan nilai mahasiswa di minggu lain.");
    mhsNilaiTertinggi(mingguTertinggi);
}
}

```

TUGAS KELOMPOK

Implementasikan fungsi ke dalam project kelompok Anda. Jangan lupa, semoga kode program harus di-push ke repository Anda

Jawab:

Untuk implementasi kelompok kami telah mengaplikasikan fungsi di project kami, berikut adalah link repository projek kami

https://github.com/tora180405/Sistem_Akademik