

LAPORAN
PRAKTIKUM DASPRO PERTEMUAN 13



Nama :

Ahmad Dzul Fadhli Hanna

NIM :

2341720106

Kelas :

TI-1H

Absen :

03

Tujuan :

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.

Praktikum

2.1 Percobaan 1

Praktikum : Membuat Fungsi Tanpa Parameter

Codingan

```
public class TerimaKasih_03 {  
  
    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) {  
        UcapanTerimakasih();  
    }  
}
```

Hasil Running

```
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> javac TerimaKasih_03.java  
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> java TerimaKasih_03  
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.  
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> |
```

Pertanyaan

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

Tidak, fungsi tanpa parameter tidak selalu digunakan untuk fungsi "void" saja. Fungsi tanpa parameter dapat mengembalikan nilai (non-void) jika didefinisikan dengan tipe data pengembalian selain void.

2. Apakah bisa kalimat "Thank you for.....dst" dituliskan tanpa fungsi UcapanTerimaKasih? modifikasi kode program tersebut untuk menampilkan suatu kalimat tanpa dan dengan menggunakan fungsi!

Ya, kalimat "Thank you for... dst" dapat dituliskan tanpa menggunakan fungsi.

Dengan menuliskan kalimat langsung pada fungsi main. Berikut modifikasi kodenya:

```
public class TerimaKasih_03 {  
    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.");  
    }  
}
```

```
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> javac TerimaKasih_03.java  
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> java TerimaKasih_03  
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.  
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> |
```

3. Apakah keuntungan menggunakan fungsi di dalam program?

- Modularitas: Fungsi memungkinkan Anda membagi program menjadi bagian-bagian kecil yang lebih mudah dikelola (modular). Ini membuat kode lebih terstruktur dan mudah dimengerti.
- Dapat digunakan ulang: Fungsi dapat dipanggil dari berbagai bagian dalam program atau bahkan dari program lain, memungkinkan penggunaan ulang kode.
- Mempermudah debugging: Jika terdapat kesalahan, debugging dapat menjadi lebih mudah karena fokus pada fungsi tertentu tanpa harus memeriksa seluruh program.
- Abstraksi: Fungsi memungkinkan abstraksi, yaitu menyembunyikan detail implementasi dan hanya mengekspos antarmuka yang diperlukan. Ini memudahkan penggunaan dan pemeliharaan kode.

- Penyusunan kode yang lebih efisien: Dengan memisahkan tugas ke dalam fungsi-fungsi kecil, Anda dapat lebih efisien dalam menulis, memelihara, dan memperbarui kode.

2.2 Percobaan 2 : Membuat Fungsi Dengan Parameter

Praktikum

Codingan

```
public class TerimaKasih02_03 {  
    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 ucapanTambahan(String ucapan) {  
        System.out.println(ucapan);  
    }  
    public static void main(String[] args) {  
        UcapanTerimakasih();  
        String ucapan = "Terima Kasih Pak.. Bu.. Semoga Sehat  
selalu";  
        ucapanTambahan(ucapan);  
    }  
}
```

Hasi Running

```
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> javac TerimaKasih02_03.java  
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> java TerimaKasih02_03  
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.  
Terima Kasih Pak.. Bu.. Semoga Sehat selalu
```

Pertanyaan

1. Apakah kegunaan suatu parameter di dalam fungsi?

- Parameter digunakan untuk memberikan input ke dalam fungsi. Fungsi dapat menerima nilai atau data melalui parameter yang didefinisikan pada saat fungsi dipanggil.
- Parameter memungkinkan fleksibilitas dalam penggunaan fungsi. Dengan menggunakan parameter, Anda dapat membuat fungsi yang dapat bekerja dengan berbagai nilai tanpa harus menulis ulang logika fungsinya.

2. Apakah parameter sama dengan variabel? jelaskan!

Ya, parameter pada dasarnya adalah variabel yang digunakan untuk menerima nilai input ketika fungsi dipanggil. Mereka bertindak sebagai wadah untuk nilai yang akan digunakan di dalam fungsi.

Perbedaan utama adalah bahwa variabel didefinisikan di dalam tubuh fungsi, sedangkan parameter didefinisikan dalam tanda kurung fungsi pada saat deklarasi. Parameter akan mengambil nilai yang diberikan saat fungsi dipanggil.

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

Tidak, parameter di Java tidak hanya digunakan untuk melewati data input, tetapi juga untuk mengembalikan data output. Programmer dapat menggunakan tipe data pengembalian pada fungsi untuk menentukan jenis data yang akan dikembalikan oleh fungsi tersebut.

2.3 Percobaan 3 : Membuat Fungsi dengan Nilai Kembalian

Praktikum

1. Codingan

```
import java.util.Scanner;

public class Ucapan_03 {

    public static String PenerimaanUcapan() {

        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 main(String[] args) {

        String nama = PenerimaanUcapan();

        System.out.println("Thank you " + nama + "\nMay the
force be with you.");

    }

}
```

Hasil running

```
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> javac Ucapan_03.java
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> java Ucapan_03
Tuliskan NAMA orang yang ingin Anda beri ucapan
Ibu
Thank you Ibu
May the force be with you.
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> █
```

Pertanyaan

1. Jelaskan mengenai, kapan suatu fungsi membutuhkan nilai kembalian (return)!
Fungsi membutuhkan nilai kembalian (return) ketika hasil dari eksekusi fungsi tersebut perlu dikembalikan ke bagian pemanggil fungsi. Misalnya, jika Programmer memiliki suatu perhitungan atau pemrosesan di dalam fungsi dan ingin menggunakan hasilnya di bagian lain dari program, Programmer akan menggunakan pernyataan return untuk mengembalikan nilai tersebut.
2. Apakah bisa ditambahkan statement System.out.println di dalam fungsi dengan nilai kembalian? Apa pengaruh nya?
Ya, bisa ditambahkan statement System.out.println di dalam fungsi dengan nilai kembalian. Pernyataan ini akan mencetak hasil ke konsol. Dan pengaruhnya adalah setelah statement return dieksekusi, eksekusi fungsi segera keluar dan statement apa pun setelahnya tidak akan dieksekusi.
3. Apakah bisa fungsi tanpa nilai kembalian di dalam fungsi main dipanggil tanpa dilewatkan ke variabel? seperti pada percobaan 1? Jelaskan!
Ya, fungsi tanpa nilai kembalian dapat dipanggil tanpa harus dilewatkan ke variabel. Ketika fungsi tanpa nilai kembalian dipanggil, pemanggil hanya menjalankan kode di dalam fungsi dan tidak memperoleh nilai yang dikembalikan.

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

Praktikum

Codingan

```

import java.util.Scanner;

public class UcapanTerimaKasih_03 {

    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();

        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 main(String[] args) {

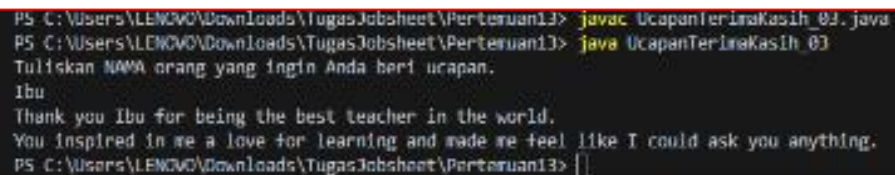
        UcapanTerimakasih();

    }

}

```

Hasil running



```

PS C:\Users\LENGVO\Downloads\TugasJobsheet\Peraturan13> javac UcapanTerimaKasih_03.java
PS C:\Users\LENGVO\Downloads\TugasJobsheet\Peraturan13> java UcapanTerimaKasih_03
Tuliskan NAMA orang yang ingin Anda beri ucapan.
Ibu
Thank you Ibu 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.
PS C:\Users\LENGVO\Downloads\TugasJobsheet\Peraturan13>

```

Pertanyaan

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

Fungsi yang akan dikerjakan pertama kali adalah fungsi UcapanTerimakasih(). Terlihat dari fungsi main(String[] args) di mana hanya fungsi UcapanTerimakasih() yang dipanggil.

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

Tidak ada aturan khusus yang mengharuskan penempatan metode di atas atau di bawah fungsi main, tetapi konsistensi dalam penempatan dapat membantu membuat kode lebih mudah dibaca dan dipahami.

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

Kode yang dimodifikasi

```
import java.util.Scanner;

public class UcapanTerimaKasih_03 {

    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 UcapanTambahan(String ucapanTambahan) {

        System.out.println("Ucapan Tambahan: " + ucapanTambahan);

    }

    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 main(String[] args) {

        UcapanTerimakasih();

        UcapanTambahan("Semoga sukses selalu!");

    }

}
```


Hasil running

```
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> javac UcapanTerimaKasih_03.java
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> java UcapanTerimaKasih_03
Tuliskan NAMA orang yang ingin Anda beri ucapan.
Ibu
Thank you Ibu 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.
Ucapan Tambahan: Semoga sukses selalu!
```


4. Silakan commit dan push ke repository Anda.

```
C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan11>git commit -m "P11 ke 3"
[main 2df1674] P11 ke 3
4 files changed, 53 insertions(+)
create mode 100644 Pertemuan11/Quiz03.class
create mode 100644 Pertemuan11/Quiz03.java
create mode 100644 Pertemuan11/Triangle03.class
create mode 100644 Pertemuan11/Triangle03.java

C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan11>git push -u origin main
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (7/7), done.
Writing objects: 100% (7/7), 2.34 KiB | 239.00 KiB/s, done.
Total 7 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/ahmaddzulfadhlihanan/TugasJobsheet.git
   7b9e452..2df1674  main -> main
branch 'main' set up to track 'origin/main'.
```

 Quiz03.java

P11 ke 3

1 minute ago

2.5 Percobaan 5: Fungsi Varargs

Praktikum

Codingan

```
public class Percobaan503 {

    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();

    }

    public static void main(String[] args) {

        Tampil("Daspro 2023", 100, 200);

        Tampil("Teknologi informasi", 1, 2, 3, 4, 5);

        Tampil("Polinema");

    }

}
```

Hasil Running

```
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> javac Percobaan503.java
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> java Percobaan503
String: Daspro 2023
Jumlah argumen/parameter: 2
100 200
String: Teknologi informasi
Jumlah argumen/parameter: 5
1 2 3 4 5
String: Polinema
Jumlah argumen/parameter: 0
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> |
```

Pertanyaan

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

Penulisan parameter dengan int... a menggunakan varargs di Java. Varargs (variable-length argument lists) memungkinkan kita untuk memberikan jumlah parameter yang fleksibel ke dalam suatu fungsi. Dengan menggunakan varargs, kita dapat mengirimkan nol hingga banyak argumen ke fungsi tanpa harus menuliskan overloading fungsi dengan jumlah parameter yang berbeda.

2. Sebutkan kegunaan varargs dalam implementasi kode program untuk menyelesaikan permasalahan dalam dunia nyata! (minimal 3)
 - Fleksibilitas Jumlah Argumen: Varargs memungkinkan kita untuk memiliki jumlah argumen yang fleksibel. Ini sangat bermanfaat ketika kita tidak tahu persis berapa banyak argumen yang dibutuhkan oleh suatu fungsi.
 - Menghindari Overloading Berlebihan: Tanpa varargs, kita mungkin perlu membuat beberapa versi fungsi dengan jumlah parameter yang berbeda (overloading). Varargs mengurangi kebutuhan untuk membuat banyak overloading fungsi.
 - Implementasi Kode yang Lebih Singkat: Varargs dapat membuat kode lebih singkat dan mudah dibaca, terutama jika kita memiliki fungsi yang memerlukan banyak argumen opsional.
 - Penggunaan Dua Tipe Data Varargs dalam Satu Fungsi: Kita dapat menggunakan dua tipe data varargs dalam satu fungsi. Namun, setiap tipe data varargs harus didefinisikan dalam satu parameter yang berbeda.

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

Ya, kita dapat menggunakan dua tipe data varargs dalam satu fungsi.

Namun, setiap tipe data varargs harus didefinisikan dalam satu parameter yang berbeda.

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

```
import java.util.Scanner;

public class Percobaan603 {

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

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

    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 adalah " + L);

        vol = hitungVolume(t, p, l);
        System.out.println("Volume balok adalah " + vol);
        input.close();
    }
}
```

Hasil Running / Validasi Hasil

```

PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> javac Percobaan603.java
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> java Percobaan603
Masukkan panjang
2
Masukkan lebar
1
Masukkan tinggi
3
Luas persegi adalah 2
Volume balok adalah 6

```

Pertanyaan

1. Sebutkan tahapan dan urutan eksekusi pada percobaan 6 di atas!
 - Deklarasi Variabel dan Objek Scanner
 - Input Panjang, Lebar, dan Tinggi
 - Pemanggilan Fungsi hitungLuas
 - Memanggil fungsi hitungLuas(p, l)
 - Menghitung luas persegi dengan panjang (p) dan lebar (l) yang telah diinput.
 - Menyimpan hasil perhitungan dalam variabel L.
 - Output Luas Persegi
 - Pemanggilan Fungsi hitungVolume
 - Memanggil fungsi hitungVolume(t, p, l)
 - Menghitung volume balok dengan tinggi (t), panjang (p), dan lebar (l) yang telah diinput.
 - memanggil fungsi hitungLuas(a, b) untuk mendapatkan luas persegi dengan panjang (a) dan lebar (b).
 - Menyimpan hasil perhitungan dalam variabel vol.
 - Output Volume Balok
 - Menutup objek Scanner
2. 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 }

```

Output

```

PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> javac Programku03.java
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> java Programku03
1234567
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13>

```

Alur

- Pada main, temp diinisialisasi dengan hasil penjumlahan dari fungsi Jumlah yaitu 2.
 - Kemudian, fungsi TampilJumlah dipanggil dengan parameter temp dan 5.
 - Pada TampilJumlah, fungsi Jumlah dipanggil dan mengembalikan nilai 7.
 - Selanjutnya, hasil penjumlahan 7 dikirim sebagai parameter ke fungsi TampilHinggaKei.
 - Di dalam TampilHinggaKei, dilakukan perulangan untuk mencetak angka dari 1 hingga 7 tanpa spasi.
 - Akhirnya, output yang dihasilkan adalah 12345.
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!

Penggunaan Parameter:

Ketika Fungsi Memerlukan Input Eksternal: Fungsi menggunakan parameter ketika memerlukan nilai atau data eksternal sebagai masukan untuk melakukan operasi tertentu. Parameter memberikan cara untuk mentransfer nilai ke dalam fungsi.

Ketika Ada Ketergantungan Antara Fungsi: Jika ada hubungan atau ketergantungan antara beberapa fungsi, parameter dapat digunakan untuk mentransfer informasi antar mereka.

Penggunaan Nilai Kembalian:

Ketika Fungsi Menghasilkan Hasil yang Diperlukan: Fungsi menggunakan nilai kembalian ketika menghasilkan suatu nilai atau hasil yang diperlukan oleh pemanggilnya. Misalnya, menghitung suatu nilai, menyaring data, atau melakukan operasi matematis.

Ketika Ada Kebutuhan Mengembalikan Informasi: Jika ada informasi yang perlu dikembalikan dari fungsi untuk digunakan di tempat lain dalam program.

Tugas Mandiri dan Kelompok

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

Codingan

```
import java.util.Scanner;

public class Kubus03 {

    static int hitungVolume(int s) {

        int volume = s * s * s;

        return volume;

    }

    static int hitungLuasPermukaan(int rusuk) {

        int luasPermukaan = rusuk * rusuk * 6;

        return luasPermukaan;

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        int r, vol, lPermukaan;

        System.out.println("Masukkan panjang sisi / rusuk");

        r = sc.nextInt();

        vol = hitungVolume(r);

        System.out.println("Volume kubus adalah " + vol);

        lPermukaan = hitungLuasPermukaan(r);

        System.out.println("Luas permukaan kubus adalah " +
lPermukaan);

        sc.close();

    }

}
```

Hasil running

```
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> javac Kubus03.java
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> java Kubus03
Masukkan panjang sisi / rusuk
2
Volume kubus adalah 8
Luas permukaan kubus adalah 24
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> 
```


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-)

Codingan

```

import java.util.Scanner;

public class NilaiLimaMahasiswa03 {

    // fungsi input data nilai
    public static int[][] inputData() {
        Scanner scan = new Scanner(System.in);
        int[][] nilai = new int[7][5];
        String[] nama = { "Sari", "Rina", "Yani", "Dwi",
"Lusi" };
        for (int i = 0; i < 7; i++) {
            System.out.println("Minggu ke-" + (i + 1) + ":
");
            for (int j = 0; j < 5; j++) {
                System.out.print("Nilai mahasiswa " +
nama[j] + " :");
                nilai[i][j] = scan.nextInt();
            }
        }
        scan.close();
        return nilai;
    }

    // fungsi menampilkan seluruh nilai mahasiswa dari
    minggu 1 sampai 7
    public static void tampilData(int[][] nilai) {
        String[] nama = { "Sari", "Rina", "Yani", "Dwi",
"Lusi" };
        for (int i = 0; i < 5; i++) {
            System.out.print(nama[i] + "\t");
            for (int j = 0; j < 7; j++) {
                System.out.print(nilai[j][i] + "\t");
            }
            System.out.println();
        }
    }
}

```

```

        // fungsi pada minggu keberapa terdapat nilai tertinggi
dibanding minggu lain

        // dari data nilai keseluruhan

        public static int cariMingguNilaiTertinggi(int[][]
nilai) {

            int minggu = 0;

            int total = 0;

            int max = 0;

            for (int j = 0; j < 7; j++) {

                total = 0;

                for (int i = 0; i < 5; i++) {

                    total += nilai[j][i];

                }

                if (total > max) {

                    max = total;

                    minggu = j + 1;

                }

            }

            return minggu;

        }

        // fungsi mahasiswa yang memiliki nilai tertinggi dengan
keterangan minggu

        // keberapa

        public static void tampilMahasiswaNilaiTertinggi(int[][]
nilai) {

            String[] nama = { "Sari", "Rina", "Yani", "Dwi",
"Lusi" };

            int max = 0;

            int minggu = 0;

            for (int i = 0; i < 7; i++) {

                for (int j = 0; j < 5; j++) {

                    if (nilai[i][j] > max) {

                        max = nilai[i][j];

                        minggu = i + 1;

                    }

                }

            }

        }

```

```

        }

    }

    for (int i = 0; i < 7; i++) {
        for (int j = 0; j < 5; j++) {
            if (nilai[i][j] == max) {
                System.out.println(nama[j] + "\t:\t" +
nilai[i][j] + "\t" + "Pada minggu ke-" + minggu);
            }
        }
    }

}

// Console
public static void main(String[] args) {
    int[][] nilai = inputData();
    tampilData(nilai);
    int minggu = cariMingguNilaiTertinggi(nilai);
    System.out.println("Minggu dengan nilai tertinggi
adalah minggu ke-" + minggu);
    tampilMahasiswaNilaiTertinggi(nilai);
}
}

```

Hasil Running

```
PS C:\Users\wearnes\Downloads\TugasJobsheet\Pertemuan13> java NilailimaMahasiswa03
Minggu ke-1:
Nilai mahasiswa Sari:11
Nilai mahasiswa Rina:12
Nilai mahasiswa Yani:13
Nilai mahasiswa Dwi:14
Nilai mahasiswa Lusi:15
Minggu ke-2:
Nilai mahasiswa Sari:16
Nilai mahasiswa Rina:1
Nilai mahasiswa Yani:17
Nilai mahasiswa Dwi:18
Nilai mahasiswa Lusi:19
Minggu ke-3:
Nilai mahasiswa Sari:15
Nilai mahasiswa Rina:20
Nilai mahasiswa Yani:21
Nilai mahasiswa Dwi:22
Nilai mahasiswa Lusi:23
Minggu ke-4:
Nilai mahasiswa Sari:24
Nilai mahasiswa Rina:25
Nilai mahasiswa Yani:26
Nilai mahasiswa Dwi:27
Nilai mahasiswa Lusi:28
Minggu ke-5:
Nilai mahasiswa Sari:29
Nilai mahasiswa Rina:30
Nilai mahasiswa Yani:31
Nilai mahasiswa Dwi:32
Nilai mahasiswa Lusi:33
Minggu ke-6:
Nilai mahasiswa Sari:34
Nilai mahasiswa Rina:35
Nilai mahasiswa Yani:36
Nilai mahasiswa Dwi:37
Nilai mahasiswa Lusi:38
Minggu ke-7:
Nilai mahasiswa Sari:39
Nilai mahasiswa Rina:40
Nilai mahasiswa Yani:41
Nilai mahasiswa Dwi:42
Nilai mahasiswa Lusi:43
Sari    11    16    15    24    29    34    39
Rina    12     1    20    25    30    35    40
Yani    13    17    21    26    31    36    41
Dwi     14    18    22    27    32    37    42
Lusi    15    19    23    28    33    38    43
Minggu dengan nilai tertinggi adalah minggu ke-7
Lusi :      43      Pada minggu ke-7
PS C:\Users\wearnes\Downloads\TugasJobsheet\Pertemuan13>
```

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

Codingan

```

import java.util.Scanner;

public class NilaiMahasiswa03 {
    static Scanner scanner = new Scanner(System.in);

    public static void main(String[] args) {
        // Input Jumlah Mahasiswa dan Jumlah Minggu
        System.out.print("Masukkan jumlah mahasiswa: ");
        int jumlahMahasiswa = scanner.nextInt();

        System.out.print("Masukkan jumlah minggu: ");
        int jumlahMinggu = scanner.nextInt();

        // Inisialisasi Array 2 Dimensi
        int[][] nilaiMahasiswa = new
int[jumlahMahasiswa][jumlahMinggu];

        // Input Nilai Minggu
        for (int j = 0; j < jumlahMinggu; j++) {
            System.out.print("Masukkan nilai untuk minggu ke-"
+ (j + 1) + ": \n");

            for (int i = 0; i < jumlahMahasiswa; i++) {
                System.out.print("Nilai mahasiswa ke-" + (i +
1) + ": ");

                nilaiMahasiswa[i][j] = scanner.nextInt();
            }
        }

        tampilkanNilaiMahasiswa(nilaiMahasiswa,
jumlahMahasiswa, jumlahMinggu);

        // Cari Minggu dengan Nilai Tertinggi
        int mingguTertinggi =
cariMingguTertinggi(nilaiMahasiswa, jumlahMahasiswa,
jumlahMinggu);

        System.out.println("Minggu ke-" + mingguTertinggi + "
memiliki nilai tertinggi.");

        // Tampilkan Mahasiswa dengan Nilai Tertinggi
        tampilkanMahasiswaTertinggi(nilaiMahasiswa,
jumlahMahasiswa, jumlahMinggu);
    }

    // Fungsi untuk input nilai mahasiswa
    public static void inputNilaiMahasiswa(int[][]
nilaiMahasiswa, int minggu, int jumlahMahasiswa) {
        System.out.println("Masukkan nilai untuk minggu ke-" +
minggu + ": ");

        for (int i = 0; i < jumlahMahasiswa; i++) {

```

```

        System.out.print("Nilai mahasiswa ke-" + (i + 1) +
": ");

        nilaiMahasiswa[i][minggu] = scanner.nextInt();

    }

}

// Fungsi untuk menampilkan nilai mahasiswa
public static void tampilkanNilaiMahasiswa(int[][]
nilaiMahasiswa, int jumlahMahasiswa, int jumlahMinggu) {

    System.out.println("\nNilai Mahasiswa:");

    for (int i = 0; i < jumlahMahasiswa; i++) {

        System.out.print("Mahasiswa ke-" + (i + 1) + ": ");

        for (int j = 0; j < jumlahMinggu; j++) {

            System.out.print(nilaiMahasiswa[i][j] + " ");

        }

        System.out.println();

    }

}

// Fungsi untuk mencari minggu keberapakah terdapat nilai
tertinggi

public static int cariMingguTertinggi(int[][]
nilaiMahasiswa, int jumlahMahasiswa, int jumlahMinggu) {

    int mingguTertinggi = 1;

    int nilaiTertinggi = nilaiMahasiswa[0][0];

    for (int j = 0; j < jumlahMinggu; j++) {

        int totalNilaiHari = 0;

        for (int i = 0; i < jumlahMahasiswa; i++) {

            totalNilaiHari += nilaiMahasiswa[i][j];

        }

        if (totalNilaiHari > nilaiTertinggi) {

            nilaiTertinggi = totalNilaiHari;

            mingguTertinggi = j + 1;

        }

    }

    return mingguTertinggi;

}

// Fungsi untuk menampilkan mahasiswa dengan nilai
tertinggi untuk setiap minggu

public static void tampilkanMahasiswaTertinggi(int[][]

```

```

nilaiMahasiswa, int jumlahMahasiswa, int jumlahMinggu) {

    System.out.println("\nMahasiswa dengan Nilai Tertinggi
untuk Setiap Minggu:");

    for (int j = 0; j < jumlahMinggu; j++) {

        int nilaiTertinggi = 0;

        int mahasiswaTertinggi = 0;

        for (int i = 0; i < jumlahMahasiswa; i++) {

            if (nilaiMahasiswa[i][j] > nilaiTertinggi) {

                nilaiTertinggi = nilaiMahasiswa[i][j];

                mahasiswaTertinggi = i + 1;

            }

        }

        System.out.println("Minggu ke-" + (j + 1) + ":
Mahasiswa ke-" + mahasiswaTertinggi);

    }

}

}

```

Hasil running

```

PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> javac NilaiMahasiswa03.java
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13> java NilaiMahasiswa03
Masukkan jumlah mahasiswa: 3
Masukkan jumlah minggu: 2
Masukkan nilai untuk minggu ke-1:
Nilai mahasiswa ke-1: 78
Nilai mahasiswa ke-2: 75
Nilai mahasiswa ke-3: 81
Masukkan nilai untuk minggu ke-2:
Nilai mahasiswa ke-1: 75
Nilai mahasiswa ke-2: 76
Nilai mahasiswa ke-3: 74

Nilai Mahasiswa:
Mahasiswa ke-1: 78 75
Mahasiswa ke-2: 75 76
Mahasiswa ke-3: 81 74
Minggu ke-1 memiliki nilai tertinggi.

Mahasiswa dengan Nilai Tertinggi untuk Setiap Minggu:
Minggu ke-1: Mahasiswa ke-3
Minggu ke-2: Mahasiswa ke-2
PS C:\Users\LENOVO\Downloads\TugasJobsheet\Pertemuan13>

```


Tugas Projek

Codingan

```
import java.util.Scanner;

public class FiturMengaturDiskon {
    public static double Diskon(double totalPembelian) {
        double potonganDiskon;

        if (totalPembelian >= 300000) {
            potonganDiskon = totalPembelian * 0.15;
        } else if (100000 <= totalPembelian) {
            potonganDiskon = totalPembelian * 0.10;
        } else if (60000 <= totalPembelian) {
            potonganDiskon = totalPembelian * 0.05;
        } else {
            potonganDiskon = 0;
        }

        return potonganDiskon;
    }

    public static void main(String[] args) {
        // Scanner
        Scanner sc = new Scanner(System.in);

        // Variabel
        double totalHarga, bayar;

        // Input total pembelian
        System.out.print("Total Pembelian : ");
        totalHarga = sc.nextDouble();

        // Ketentuan diskon
        double potonganDiskon = Diskon(totalHarga);
```

```

        // Bayar
        bayar = totalHarga - potonganDiskon;

        // Riwayat diskon
        System.out.println("Potongan Diskon : " +
potonganDiskon);

        System.out.println("Bayar          : " + bayar);

        sc.close();
    }
}

        // Riwayat diskon
        System.out.println("Potongan Diskon : " +
potonganDiskon);

        System.out.println("Bayar          : " + bayar);

        sc.close();
    }
}

```


Hasil Running

```

PS C:\Users\wearnes\Downloads\Kasir-Salon> & 'C:\Program Files\Java\jdk-20\bin\java.exe' '--enable-preview' '-XX:+
ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\wearnes\AppData\Roaming\Code\User\workspaceStorage\36ffad401513
b2bdf74e5ea32dd7430d\redhat.java\jdt_ws\Kasir-Salon_797e6781\bin' 'FiturMengaturDiskon'
Total Pembelian : 450000
Potongan Diskon : 67500.0
Bayar          : 382500.0
PS C:\Users\wearnes\Downloads\Kasir-Salon> 

```

Commit

 FiturMengaturDiskon.java

evan punya

14 minutes ago