

LAPORAN PRAKTIKUM DASAR PEMROGRAMAN

JOBSHEET 13: FUNGSI 1



Nama: Afifah Khoirunnisa

NIM: 2341720250

Kelas: 1B

Prodi : D-IV Teknik Informatika

2. Praktikum 2.1

Percobaan 1: Membuat Fungsi Tanpa Parameter

Waktu Percobaan: 40 menit

Source code:

```
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.");
    }

    Run | Debug
    public static void main (String[] args) {
        UcapanTerimaKasih();
    }
}
```

Output:

```
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 D:\COLLEGE\SEMESTER 1\PRAKTIKUM DASAR PEMROGRAMAN\PRAKTIKUM\WEEK 13>
```

Pertanyaan!

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

Jawaban:

Tidak, fungsi tanpa parameter tidak selalu digunakan untuk fungsi “void” saja. Fungsi tanpa parameter juga dapat digunakan untuk fungsi yang memiliki nilai balik.

- 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!

Jawaban:

Kalimat “Thank you for...dst” dapat dituliskan tanpa fungsi UcapanTerimaKasih.

Source code:

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

    Run | Debug
    public static void main (String[] args) {
        UcapanTerimaKasih();

        //Tanpa 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.");
    }
}
```

Output:

```
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.
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 D:\COLLEGE\SEMESTER 1\PRAKTIKUM DASAR PEMROGRAMAN\PRAKTIKUM\WEEK 13>
```

- 3. Apakah keuntungan menggunakan fungsi di dalam program?

Jawaban:

Ada beberapa keuntungan menggunakan fungsi di dalam program, antara lain:

- Kerja program menjadi lebih terstruktur. Fungsi dapat digunakan untuk membagi program menjadi beberapa bagian yang lebih kecil dan lebih mudah dipahami.
- Kode program menjadi lebih mudah diuji. Fungsi dapat diuji secara independen, sehingga memudahkan dalam menemukan dan memperbaiki kesalahan (bug) pada program.
- Kode program menjadi lebih mudah untuk dimaintain. Fungsi dapat diubah atau ditambahkan tanpa mempengaruhi bagian lain dari program.
- Kode program menjadi lebih efisien. Fungsi dapat digunakan untuk melakukan tugas yang berulang-ulang, sehingga mengurangi jumlah kode yang perlu ditulis.

2.2 Percobaan 2: Membuat Fungsi Dengan Parameter

Waktu Percobaan: 40 menit

Source code:

```
public class TerimaKasih_03 {  
    //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 UcapanTambahan (String ucapan) {  
        System.out.println(ucapan);  
    }  
  
    Run | Debug  
    public static void main (String[] args) {  
        UcapanTerimaKasih();  
        String ucapan = "Terima KASih Pak.. Bu..Semoga Sehat Selalu";  
        UcapanTambahan(ucapan);  
    }  
}
```

Output:

```
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  
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM DASAR PEMROGRAMAN\PRAKTIKUM\WEEK 13>
```

Pertanyaan!

1. Apakah kegunaan suatu parameter di dalam fungsi?

Jawaban:

Parameter adalah variable yang digunakan untuk menerima data dari luar fungsi.

Data ini dapat berupa nilai data, objek, dtau fungsi lain. Parameter digunakan untuk membuat fungsi menjadi lebih fleksibel dan mudah digunakan.

Berikut adalah beberapa kegunaan parameter di dalam fungsi:

- Untuk menerima data input dari pemanggil fungsi. Misalnya, fungsi cetak_nama() menerima parameter nama untuk menerima nama yang akan dicetak.
- Untuk mengontrol aliran program. Misalnya, fungsi periksa_usia() menerima parameter usia untuk menentukan apakah seseorang sudah cukup umur.
- Untuk berbagi data antar fungsi. Misalnya, fungsi simpan_data() dan baca_data() dapat menggunakan parameter yang sama untuk menyimpan dan membaca data dari file.

2. Apakah parameter sama dengan variabel? jelaskan!

Jawaban:

Parameter tidak sama dengan variabel. Parameter adalah variabel yang hanya didefinisikan di dalam fungsi. Variabel adalah variabel yang dapat didefinisikan di mana saja dalam program.

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

Jawaban:

Parameter di dalam Bahasa pemrograman Java dapat digunakan untuk melewati data input maupun data output. Untuk melewati data output, fungsi dapat menggunakan nilai balik (return value). Nilai balik adalah nilai yang dikembalikan oleh fungsi kepada pemanggilnya. Nilai balik ini dapat berupa nilai data, objek, atau fungsi lain.

2.3 Percobaan 3: Membuat Fungsi dengan Nilai Kembalian

Waktu Percobaan: 40 menit

Source code:

```
import java.util.Scanner;
public class Ucapan_03 {
    public static String PenerimaUcapan() {
        Scanner sc = new Scanner (System.in);
        System.out.print("Tuliskan NAMA orang yang ingin Anda beri ucapan: ");
        String namaOrang = sc.nextLine();
        sc.close();
        return namaOrang;
    }

    Run | Debug
    public static void main(String[] args) {
        String nama = PenerimaUcapan();
        System.out.println("Thank you " + nama + "\nMay the force be with you.");
    }
}
```

Output:

```
Tuliskan NAMA orang yang ingin Anda beri ucapan: Afifah
Thank you Afifah
May the force be with you.
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM DASAR PEMROGRAMAN\PRAKTIKUM\WEEK 13>
```

Pertanyaan!

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

Jawaban:

Suatu fungsi membutuhkan nilai kembalian (return) jika fungsi tersebut menghasilkan nilai yang perlu dikembalikan kepada pemanggil fungsi. Nilai balik ini dapat berupa nilai data, objek, atau fungsi lain. Jika suatu fungsi tidak menghasilkan nilai yang perlu dikembalikan, maka fungsi tersebut dapat menggunakan tipe void.

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

Jawaban:

`System.out.println` dapat ditambahkan di dalam fungsi dengan nilai kembalian. Statement ini akan mencetak output ke layar.

Pengaruh dari menambahkan statement `System.out.println` di dalam fungsi dengan nilai kembalian adalah:

- Output dari statement `System.out.println` akan ditampilkan ke layar.
- Nilai balik dari fungsi tetap akan dikembalikan kepada pemanggil fungsi.

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

Jawaban:

Ya, fungsi tanpa nilai kembalian di dalam fungsi main dapat dipanggil tanpa dilewati ke variabel. Hal ini karena fungsi tanpa nilai kembalian tidak menghasilkan nilai yang perlu dikembalikan kepada pemanggil fungsi.

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

Waktu Percobaan: 50 menit

Source code:

```
import java.util.Scanner;

public class UcapanTerimaKasih_03 {
    public static String PenerimaUcapan() {
        Scanner sc = new Scanner(System.in);
        System.out.println(x:"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 te world.\n" +
            "You inspired in me a love for learning and made me feel like I could askk you anything.");
    }

    Run | Debug
    public static void main(String[] args) {
        UcapanTerimaKasih();
    }
}
```

Output:

```
Tuliskan NAMA orang yang ingin Anda beri ucapan:
Bu Triana
Thank you Bu Triana for being the best teacher in te world.
You inspired in me a love for learning and made me feel like I could askk you anything.
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM DASAR PEMROGRAMAN\PRAKTIKUM\WEEK 13> █
```

Pertanyaan!

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

Jawaban:

Berdasarkan uji coba pada percobaan 4, fungsi yang akan dieksekusi pertama kali adalah fungsi main(). Hal ini karena fungsi main() adalah fungsi utama dari program. Fungsi main() akan dieksekusi pertama kali ketika program dijalankan.

Fungsi PenerimaUcapan() akan dieksekusi setelah fungsi main() dieksekusi. Hal ini karena fungsi PenerimaUcapan() dipanggil oleh fungsi main().

Berikut adalah urutan eksekusi fungsi pada percobaan 4:

- Fungsi main()
- Fungsi PenerimaUcapan()

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

Jawaban:

Tata cara penulisan fungsi yang benar di dalam class adalah di atas fungsi main. Hal ini karena fungsi main() adalah fungsi utama dari program. Fungsi harus ditulis terlebih dahulu agar dapat memanggil fungsi-fungsi lain yang ada di dalam 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.

Jawaban:

Source code:

```
import java.util.Scanner;

public class UcapanTerimaKasih_03 {
    public static String PenerimaUcapan() {
        Scanner sc = new Scanner(System.in);
        System.out.println(x:"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 te world.\n" +
            "You inspired in me a love for learning and made me feel like I could askk you anything.");
    }

    public static void UcapanTambahkan(String ucapanTambahkan) {
        System.out.println(ucapanTambahkan);
    }

    public static void main(String[] args) {
        UcapanTerimaKasih();
        UcapanTambahkan(ucapanTambahkan:"You are always there for me, no matter what.");
    }
}
```

Output:

```
Tuliskan NAMA orang yang ingin Anda beri ucapan:
Bu Triana
Thank you Bu Triana for being the best teacher in te world.
You inspired in me a love for learning and made me feel like I could askk you anything.
You are always there for me, no matter what.
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM DASAR PEMROGRAMAN\PRAKTIKUM\WEEK 13>
```

2.5 Percobaan 5: Fungsi Varargs

Waktu Percobaan: 40 menit

Source code:

```
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();  
    }  
  
    Run | Debug  
    public static void main(String[] args) {  
        Tampil(str:"Daspro 2019", ...a:100, 200);  
        Tampil(str:"Teknologi Informasi", ...a:1, 2, 3, 4, 5);  
        Tampil(str:"Polinema");  
    }  
}
```

Output:

```
String: Daspro 2019  
Jumlah argumen/parameter: 2  
100 200  
String: Teknologi Informasi  
Jumlah argumen/parameter: 5  
1 2 3 4 5  
String: Polinema  
Jumlah argumen/parameter: 0
```

Pertanyaan!

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

Jawaban:

Penulisan parameter di praktikum 5 di tulis dengan int... a karena parameter tersebut adalah varargs. Varargs adalah parameter yang dapat menerima jumlah argumen yang tidak ditentukan.

Pada praktikum 5, fungsi Tampil() menerima dua parameter. Parameter pertama adalah String yang digunakan untuk menerima string. Parameter kedua adalah int... yang digunakan untuk menerima angka.

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

Jawaban:

Berikut adalah beberapa kegunaan varargs dalam implementasi kode program untuk menyelesaikan permasalahan dalam dunia nyata:

- Untuk menerima jumlah argumen yang tidak ditentukan. Misalnya, fungsi Tampil() pada praktikum 5 dapat digunakan untuk menampilkan string dan angka apa pun, tanpa perlu dibatasi jumlah argumennya.
- Untuk membuat kode program lebih ringkas. Misalnya, fungsi Tampil() pada praktikum 5 dapat digunakan untuk menampilkan string dan angka dalam satu fungsi, tanpa perlu membuat fungsi terpisah untuk setiap jumlah argumen.
- Untuk membuat kode program lebih fleksibel. Misalnya, fungsi Tampil() pada praktikum 5 dapat digunakan untuk menampilkan string dan angka dengan jumlah argumen yang berbeda-beda, sesuai kebutuhan.

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

Jawaban:

Dalam Java, dapat menggunakan varargs (variadic arguments) untuk memungkinkan fungsi menerima jumlah argumen yang tidak terbatas dari jenis data tertentu. Namun, secara langsung, tidak dapat memiliki dua varargs parameter dalam satu fungsi. Varargs sendiri sudah mewakili sejumlah argumen yang dapat berubah dinamis, sehingga tidak perlu lebih dari satu.

```
public class ContohVarargs {
    // Fungsi dengan varargs dari tipe data int
    public static void fungsiVarargsInt(int... angka) {
        System.out.print(s:"Angka: ");
        for (int i : angka) {
            System.out.print(i + " ");
        }
        System.out.println();
    }

    // Fungsi dengan varargs dari tipe data String
    public static void fungsiVarargsString(String... kata) {
        System.out.print(s:"Kata: ");
        for (String s : kata) {
            System.out.print(s + " ");
        }
        System.out.println();
    }

    Run | Debug
    public static void main(String[] args) {
        // Memanggil fungsi dengan varargs tipe data int
        fungsiVarargsInt(...angka:1, 2, 3, 4, 5);

        // Memanggil fungsi dengan varargs tipe data String
        fungsiVarargsString(...kata:"Halo", "Ini", "Contoh", "Varargs");
    }
}
```

2.6 Percobaan 6: Pembuatan Kode Program, Dengan Fungsi versus Tanpa Fungsi Waktu Percobaan: 50 menit

Source code:

```
import java.util.Scanner;

public class Percobaan603 {
    Run | Debug
    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);

        int p,l,t,L,vol;

        System.out.println(x:"Masukkan panjang");
        p = input.nextInt();
        System.out.println(x:"Masukkan lebar");
        l = input.nextInt();
        System.out.println(x:"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);
    }

    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;
    }
}
```

Output:

```
Masukkan panjang
20
Masukkan lebar
4
Masukkan tinggi
5
Luas Persegi Panjang adalah 80
Volume balok adalah 400
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM DASAR PEMROGRAMAN\PRAKTIKUM\WEEK 13>
```

Pertanyaan!

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

Jawaban:

- a. Deklarasi Kelas dan Fungsi:
 - Deklarasi kelas Percobaan603.
 - Deklarasi fungsi main, hitungLuas, dan hitungVolume.
- b. Deklarasi Variabel:
 - Scanner input dan variabel p, l, t, L, dan vol dideklarasikan.
- c. Inisialisasi Scanner:
 - Objek Scanner dibuat untuk membaca input dari pengguna.
- d. Input Panjang, Lebar, dan Tinggi:
 - Mencetak pesan untuk meminta input panjang, lebar, dan tinggi.
 - Membaca input panjang, lebar, dan tinggi dari pengguna.
- e. Panggil Fungsi hitungLuas:
 - Memanggil fungsi hitungLuas dengan parameter p dan l untuk menghitung luas persegi panjang.
 - Hasilnya disimpan dalam variabel L.
- f. Output Luas Persegi Panjang:
 - Mencetak hasil perhitungan luas persegi panjang.
- g. Panggil Fungsi hitungVolume:
- h. Memanggil fungsi hitungVolume dengan parameter t, p, dan l untuk menghitung volume balok.
 - Hasilnya disimpan dalam variabel vol.
- i. Output Volume Balok:
 - Mencetak hasil perhitungan volume balok.

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 }
```

Jawaban:

Output:

```
1234567
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM DASAR PEMROGRAMAN\PRAKTIKUM\WEEK 13>
```

Program tersebut adalah program sederhana dalam bahasa pemrograman Java yang mendefinisikan sebuah kelas bernama `ProgramKu`. Program ini memiliki tiga metode statis: `TampilHinggaKei`, `Jumlah`, dan `TampilJumlah`. Metode `TampilHinggaKei` digunakan untuk mencetak angka dari 1 hingga nilai parameter yang diberikan tanpa spasi di antaranya. Metode `Jumlah` mengembalikan hasil penjumlahan dari dua bilangan yang diberikan. Metode `TampilJumlah` memanggil `Jumlah` untuk mendapatkan hasil penjumlahan dan kemudian memanggil `TampilHinggaKei` untuk mencetak angka dari 1 hingga hasil penjumlahan tersebut. Metode `main` dijalankan saat program dijalankan, di mana sebuah variabel `temp` diinisialisasi dengan hasil penjumlahan 1 dan 1, dan kemudian metode `TampilJumlah` dipanggil dengan parameter `temp` dan 5. Akhirnya, output program adalah angka 12, yang merupakan hasil dari penjumlahan 1+1, diikuti dengan angka dari 1 hingga 2 (hasil dari pemanggilan `TampilHinggaKei`).

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!

Jawaban:

Keputusan untuk menggunakan parameter atau nilai kembalian dalam fungsi tergantung pada sifat dan kebutuhan spesifik fungsi dalam konteks program. Parameter digunakan ketika fungsi memerlukan input eksternal atau data dari luar untuk diolah di dalamnya, meningkatkan fleksibilitas dan modularitas fungsi. Dengan menggunakan parameter, fungsi dapat diimplementasikan untuk beroperasi dengan nilai yang berbeda, memungkinkan penggunaannya dalam berbagai konteks.

Sebaliknya, nilai kembalian diperlukan ketika fungsi dimaksudkan untuk menghasilkan nilai atau informasi yang dapat digunakan di tempat lain dalam program. Hal ini relevan ketika suatu fungsi melakukan proses dan hasilnya perlu diketahui atau digunakan oleh bagian lain dari program. Fungsi void tanpa nilai kembalian biasanya cocok untuk tugas yang tidak menghasilkan output yang harus digunakan di tempat lain dalam program, seperti mencetak ke layar. Pilihan untuk menggunakan parameter, nilai kembalian, atau kombinasi keduanya didesain untuk mendukung prinsip modularitas, reusabilitas, dan pemahaman yang baik terhadap struktur dan tujuan program secara keseluruhan.

Tugas 1

Source code:

```
public class Kubus03 {
    Run | Debug
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        System.out.print(s:"Masukkan panjang sisi kubus: ");
        int sisi = sc.nextInt();

        int Vol = hitungVolume(sisi);
        System.out.println("Volume kubus tersebut adalah " + Vol + " cm3");
        int LuasPermukaan = hitungLuasPermukaan(sisi);
        System.out.println("Luas permukaan kubus tersebut adalah " + LuasPermukaan + " cm2");
        sc.close();
    }

    static int hitungVolume(int sisi) {
        int Volume = sisi * sisi * sisi;
        return Volume;
    }

    static int hitungLuasPermukaan(int sisi) {
        int LuasPermukaan = 6 * sisi * sisi;
        return LuasPermukaan;
    }
}
```

Output:

```
Masukkan panjang sisi kubus: 4
Volume kubus tersebut adalah 64 cm3
Luas permukaan kubus tersebut adalah 96 cm2
PS D:\COLLEGE\SEMESTER 1\PRAKTIKUM DASAR PEMROGRAMAN\PRAKTIKUM\WEEK 13> █
```

Tugas 2

Source code:

```
1  import java.util.Scanner;
2
3  public class InputNilai {
4  static String[] nama = {"Sari", "Rina", "Yani", "Dwi", "Lusi"};
5  static int[][] dataMahasiswa = new int[5][7];
6
7      public static int[][] inputDataMahasiswa() {
8          Scanner sc = new Scanner(System.in);
9
10         for (int i =0; i <nama.length; i++){
11             System.out.println(nama[i]);
12             for(int j=0; j < dataMahasiswa[0].length; j++) {
13                 System.out.print("Minggu ke-" + (j+1) + " = ");
14                 dataMahasiswa[i][j]= sc.nextInt();
15             }
16
17             System.out.println();
18         }
19         return dataMahasiswa;
20     }
21
22     public static void MencariNilaiTertinggi () {
23         int NilaiTertinggi = dataMahasiswa[0][0];
24         int indeksNilaiTertinggi = 0;
25         String MahasiswaNilaiTerbaik = nama[0];
26
27         for (int i =1; i <dataMahasiswa.length; i++) {
28             for (int j = 0; j < dataMahasiswa[0].length; j++) {
29                 if (dataMahasiswa[i][j] > NilaiTertinggi) {
30                     NilaiTertinggi = dataMahasiswa[i][j];
31                     indeksNilaiTertinggi = j;
32                     MahasiswaNilaiTerbaik = nama[i];
33                 }
34             }
35         }
36
37         System.out.println("Mahasiswa dengan Nilai tertinggi: " + MahasiswaNilaiTerbaik);
38         System.out.println("Nilai tertinggi adalah " + NilaiTertinggi + " pada minggu ke- " + (indeksNilaiTertinggi + 1));
39     }
40
41     public static void TampilkanData(int[][] dataMahasiswa) {
42         System.out.println("=====");
43         System.out.println("        DATA NILAI MAHASISWA        ");
44         System.out.println("=====");
45         String[] nama = {"Sari", "Rina", "Yani", "Dwi", "Lusi"};
46         for (int i = 0; i < dataMahasiswa.length; i++) {
47             System.out.print(nama[i] + " : ");
48             for (int j = 0; j < dataMahasiswa[0].length; j++) {
49                 System.out.print(dataMahasiswa[i][j]);
50                 if (j < dataMahasiswa[0].length - 1) {
51                     System.out.print(", ");
52                 }
53             }
54             System.out.println();
55         }
56     }
57
58     public static void MahasiswaNilaiTerbaik() {
59         System.out.println();
60         System.out.println("=====");
61         MencariNilaiTertinggi();
62         System.out.println("=====");
63     }
64
65     public static void main(String[] args) {
66         int [][] dataMahasiswa = inputDataMahasiswa();
67         TampilkanData(dataMahasiswa);
68         MahasiswaNilaiTerbaik();
69     }
70 }
71
72
73
```

Output:

```
=====
        DATA NILAI MAHASISWA
=====
Sari : 2, 3, 4, 2, 323, 2, 2
Rina : 12, 12, 1, 221, 2, 12, 1
Yani : 21, 21, 2, 12, 12, 1, 21
Dwi : 21, 21, 21, 2, 1, 212, 1
Lusi : 21212, 121, 21, 2, 12, 12, 1

=====
Mahasiswa dengan Nilai tertinggi: Lusi
Nilai tertinggi adalah 21212 pada minggu ke- 1
=====
```

Tugas 3:

Source code

```
1  import java.util.Scanner;
2
3  public class InputNilaiCustom {
4  static String[] nama;
5  static int[][] dataMahasiswa;
6
7  public static void inputDataMahasiswa() {
8      Scanner sc = new Scanner(System.in);
9
10     System.out.print("Masukkan jumlah Mahasiswa: ");
11     int jumlahMahasiswa = sc.nextInt();
12     System.out.println("Masukkan jumlah tugas: ");
13     int jumlahTugas = sc.nextInt();
14
15     dataMahasiswa = new int [jumlahMahasiswa][jumlahMahasiswa];
16     nama = new String[jumlahMahasiswa];
17
18     for (int i =0; i <jumlahMahasiswa; i++){
19         System.out.print("Masukkan nama mahasiswa ke-" + (i + 1) + ": ");
20         nama[i] = sc.next();
21
22         System.out.println("Masukkan nilai tugas untuk " + nama[i] + ": ");
23         for(int j=0; j < jumlahTugas; j++) {
24             System.out.print("Minggu ke-" + (j+1) + "= ");
25             dataMahasiswa[i][j]= sc.nextInt();
26         }
27         System.out.println();
28     }
29
30 }
31
32 public static void MencariNilaiTertinggi () {
33     int NilaiTertinggi = dataMahasiswa[0][0];
34     int indeksNilaiTertinggi = 0;
35     String MahasiswaNilaiTerbaik = nama[0];
36
37     for (int i =1; i <dataMahasiswa.length; i++) {
38         for (int j = 0; j < dataMahasiswa[0].length; j++) {
39             if (dataMahasiswa[i][j] > NilaiTertinggi) {
40                 NilaiTertinggi = dataMahasiswa[i][j];
41                 indeksNilaiTertinggi = j;
42                 MahasiswaNilaiTerbaik = nama[i];
43             }
44         }
45     }
46
47     System.out.println("Mahasiswa dengan Nilai tertinggi: " + MahasiswaNilaiTerbaik);
48     System.out.println("Nilai tertinggi adalah " + NilaiTertinggi + " pada minggu ke- " + (indeksNilaiTertinggi + 1));
49 }
50
51 public static void TampilkanData() {
52     System.out.println("=====");
53     System.out.println("        DATA NILAI MAHASISWA        ");
54     System.out.println("=====");
55
56     for (int i =0; i <dataMahasiswa.length; i++){
57         System.out.println(" Nilai " + nama[i] + ": ");
58         for(int j=0; j < dataMahasiswa[0].length; j++) {
59             System.out.print("Minggu ke-" + (j+1) + "= " + dataMahasiswa[i][j]);
60             System.out.println();
61         }
62         System.out.println();
63     }
64 }
65
66 public static void MahasiswaNilaiTerbaik() {
67     System.out.println();
68     System.out.println("=====");
69     MencariNilaiTertinggi();
70     System.out.println("=====");
71
72 }
73
74 public static void main(String[] args) {
75     inputDataMahasiswa();
76     TampilkanData();
77     MahasiswaNilaiTerbaik();
78 }
79 }
80
81
```

Output:

```
Masukkan jumlah Mahasiswa: 2
Masukkan jumlah tugas:
2
Masukkan nama mahasiswa ke-1: Afifah
Masukkan nilai tugas untuk Afifah:
Minggu ke-1= 89
Minggu ke-2= 67

Masukkan nama mahasiswa ke-2: Nisa
Masukkan nilai tugas untuk Nisa:
Minggu ke-1= 78
Minggu ke-2= 56

=====
          DATA NILAI MAHASISWA
=====
  Nilai Afifah:
Minggu ke-1= 89
Minggu ke-2= 67

  Nilai Nisa:
Minggu ke-1= 78
Minggu ke-2= 56

=====
Mahasiswa dengan Nilai tertinggi: Afifah
Nilai tertinggi adalah 89 pada minggu ke- 1
=====
```


Tugas kelompok:

1. Fungsi pencarian Barang

Source code:

```
// Fungsi untuk mencari barang
public static void cariBarang() {
    Scanner sc = new Scanner(System.in);
    System.out.print(s:"Masukkan kode atau barang yang ingin dicari: ");
    String cariBarang = sc.nextLine();
    boolean ditemukan = false;

    for (int kategori = 0; kategori < 4; kategori++) {
        String[] kodeArray = GudangRestoran[kategori * 3];
        String[] namaArray = GudangRestoran[kategori * 3 + 1];
        String[] JmlArray = GudangRestoran[kategori * 3 + 2];

        String kategoriJudul = JenisJudul(kategori);
        for (int i = 0; i < kodeArray.length; i++) {
            if (kodeArray[i].equalsIgnoreCase(cariBarang) || namaArray[i].equalsIgnoreCase(cariBarang)) {
                System.out.println("==== " + kategoriJudul + " ====");
                System.out.println(kodeArray[i] + " - " + namaArray[i] + " = " + JmlArray[i]);
                ditemukan = true;
            }
        }
    }
    if (!ditemukan) {
        System.out.println("Barang dengan kode atau nama " + cariBarang + " tidak ditemukan.");
    }
}
```

Output:

```
=====
|  SELAMAT DATANG DI SISTEM INVENTARIS  |
=====
Silahkan login !
Masukkan Username: afifah
Masukkan Password: fifi
Berhasil login!
=====
|  SELAMAT DATANG DI SISTEM INVENTARIS  |
=====
Sekarang Anda Berada di Menu: Staff

|1| input barang rusak
|2| Cari Barang
|9| Beralih Akun
|0| Keluar

Pilih Menu: 2
Masukkan kode atau barang yang ingin dicari: M-001
==== DAFTAR BAHAN MINUMAN ====
M-001 - Teh = 7
=====
```

Fungsi pada pemilihan judul kategori

Source code;

```
// Fungsi untuk menampilkan judul kategori
public static String JenisJudul(int kategori) {
    switch (kategori) {
        case 0:
            return "DAFTAR IKAN";
        case 1:
            return "DAFTAR HIDANGAN LAUT";
        case 2:
            return "DAFTAR BUMBU DAPUR";
        case 3:
            return "DAFTAR BAHAN MINUMAN";
        default:
            return "";
    }
}
```

Output:

Pilih Menu: 3

DAFTAR BARANG DI RESTORAN SEAFOOD		
===== DAFTAR IKAN =====		
Kode	Nama	Jumlah
I-001	Patin	5
I-002	Kakap	8
I-003	Bawal	7
I-004	Kerapu	5
I-005	Gurame	4
I-006	Bandeng	3
===== DAFTAR HIDANGAN LAUT =====		
Kode	Nama	Jumlah
H-001	Udang	8
H-002	Cumi	7
H-003	Kepiting	5
H-004	Babycrap	7
H-005	Kerang	6
===== DAFTAR BUMBU DAPUR =====		
Kode	Nama	Jumlah
B-001	Jeruk Nipis	1
B-002	Lemon	3
B-003	Bawang Putih	5
B-004	Bawang Merah	7
B-005	Merica	4
B-006	Garam	9
B-007	Minyak Goreng	10
B-008	Saos Tiram	5
B-009	Kecap manis	6
B-010	Kecap Asin	4
B-011	Saos Tomat	7
B-012	Saos sambal	8
B-013	Jahe	11
B-014	Kunyit	7
B-015	Lengkuas	8
B-016	Serai	17
B-017	Daun jeruk	5
B-018	Daun kemangi	9
===== DAFTAR BAHAN MINUMAN =====		
Kode	Nama	Jumlah
M-001	Teh	7
M-002	Lemon	6
M-003	Air putih	10