

LAPORAN PRAKTIKUM

Pertemuan III

Prinsip Perancangan Class



Disusun Oleh :

Muhammad Iqbal Ainu Rafie

19102159

Merlinda Wibowo, S.T., M. Phil

PROGRAM STUDI S1 TEKNIK INFORMATIKA

FAKULTAS INFORMATIKA

INSTITUT TEKNOLOGI TELKOM

PURWOKERTO

2021

MODUL III

Prinsip Perancangan Class

I. TUJUAN

Mahasiswa diharapkan mampu memahami tentang perancangan kelas yang baik serta menerjemahkannya dalam bahasa pemrograman.

II. DASAR TEORI

Empat prinsip utama dalam Pemrograman Berorientasi Objek:

1. Abstraction

Abstraksi berarti menggunakan hal-hal sederhana untuk mewakili hal-hal rumit.

Perhatikan contoh sederhana berikut:

```
// Before
if(input < 0 && input >= array.length)
    System.out.println("Error!");

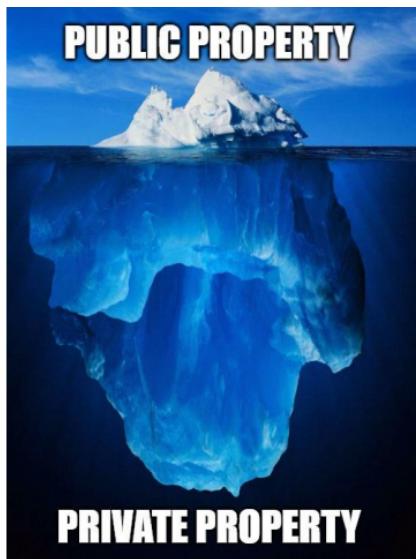
// After
boolean inputInvalid = input < 0 && input >= array.length;

if(inputInvalid)
    System.out.println("Error!");
```

Pada contoh di atas, pada if pertama, orang lain yang belum paham tentang kode kita mungkin akan bertanya-tanya kondisi apa yang sedang dicek di if tersebut. Sedangkan pada if kedua, jika kita menunjukkan kode kita pada orang awam pun, orang itu akan paham bahwa kode tersebut sedang mengecek apakah input yang diterima program valid atau invalid.

2. Encapsulation

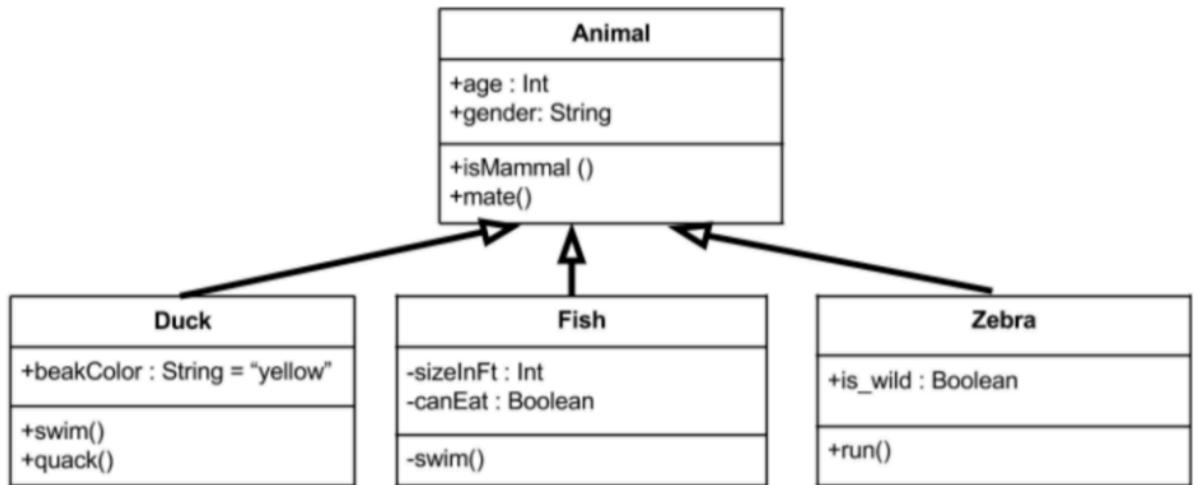
Enkapsulasi berarti menyembunyikan data-data kompleks di balik modifier Private dan menyediakan sedikit method Public sebagai “jalan” untuk mengakses data-data Private tersebut.



Bayangkan sebuah TV. Anggaplah tombol-tombol di depan TV yang dapat kita tekan adalah method Public milik TV tersebut, sedangkan proses kompleks yang terjadi di belakang layar saat kita menekan tombol-tombol tersebut adalah method dan atribut Private. Sebagai pemilik TV, kita tahu apa fungsi dari masing-masing tombol, tetapi kita tidak tahu dan tidak peduli apa proses kompleks yang terjadi di belakang layar saat kita menekan tombol tersebut.

3. Inheritance

Inheritance (arti: Pewarisan) merupakan konsep menurunkan atribut dan method milik sebuah class ke class lain. Perhatikan contoh class diagram berikut:



Pada contoh di atas, class Animal disebut Parent class atau Superclass, sedangkan class Duck, Fish, dan Zebra disebut Child class atau Subclass. Ketiga Child class tersebut akan memiliki **SEMUA** atribut dan method milik Parent class.

Dalam merancang class, jika kita menemukan ada beberapa class dengan atribut dan method yang konteksnya sama, lebih baik menggunakan konsep Pewarisan saja. Salah satu keuntungannya adalah jika kita sewaktu-waktu ingin mengubah kode kita, kita hanya perlu mengubah kode kita satu kali saja, sedangkan jika kita menuliskan class berbeda tanpa ada hubungan Inheritance, jika kita ingin melakukan perubahan kita harus merubah isi dari semua class tersebut satu persatu.

4. Polymorphism

Polimorfisme merupakan konsep pembuatan method berbeda dengan nama yang sama. Polimorfisme dibagi menjadi dua, yaitu:

a. Overloading

Overloading adalah ketika di sebuah class ada 2 atau lebih method dengan nama sama namun masing-masing memiliki parameter berbeda. Walau namanya sama, program tetap mengerti harus menjalankan method yang mana berdasarkan parameter yang dimasukan saat pemanggilan method.

Contoh sederhana method overloading:

```
● ● ●

class Hero {

    // Akan dieksekusi jika tipe data parameter adalah Armor
    public void equip(Armor armor) {}

    // Akan dieksekusi jika tipe data parameter adalah Weapon
    public void equip(Weapon weapon) {}
}
```

b. Overriding

Overriding adalah ketika sebuah Child class memiliki method dengan nama yang sama dengan Parent class-nya, namun dengan isi yang berbeda (parameter boleh sama). Nantinya program akan memprioritaskan method milik Child class ini jika ternyata ketahuan Child class memiliki method yang di-Override dari Parent-nya.

Contoh sederhana method overriding:

```
● ● ●

class Burung { // Parent class
/*
    Jika method ini dipanggil dari object Burung, method
    yang ini yang akan dieksekusi ketika terbang() dipanggil
*/
public void terbang() {
    System.out.println("Aku terbang!");
}

class Penguin extends Burung { // Child class
/*
    Jika method ini dipanggil dari object Penguin, method
    yang ini yang akan dieksekusi ketika terbang() dipanggil
*/
public void terbang() {
    System.out.println("Aku gak bisa terbang :(");
}
```

(Inheritance dan Polymorphism akan dibahas lebih lanjut di modul selanjutnya)

Beberapa prinsip-prinsip perancangan kelas yang perlu diperhatikan, antara lain:

1. Constructor. Method ini digunakan untuk inisialisasi atau mempersiapkan data untuk objek.
2. Visibilitas Bagi Atribut dan Method. Diperlukan kontrol akses untuk mengatur siapa saja yang dapat mengakses atau mengubah nilai dari atribut atau method dalam kelas tersebut sehingga penyalahgunaan atribut atau method dapat dihindari. Kontrol akses yang sering digunakan dalam perancangan suatu kelas sudah ada pada modul minggu lalu.
3. Fungsi Accessor dan Mutator. Fungsi accessor merupakan fungsi untuk mendapatkan properti dari suatu objek dan mengembalikan nilai atau value dari suatu atribut (get). Sedangkan fungsi mutator mengubah properti dari suatu objek dan mengubah nilai atau value dari sebuah atribut (set).

4. Method dan Operator Overloading. Method-method tersebut harus dapat dibedakan antara satu dengan yang lain dalam jumlah dan atau tipe datanya.
5. Melewatkkan Argumen/Parameter ke Method. Cara melewatkkan argumen/parameter ini dapat dilakukan dengan melewatkkan secara nilai (pass by value) dan melewatkkan secara referensi (pass by reference).

```

● ● ●

// VALUE
int x = 7;
int y = x;

System.out.println(x); // Output: 7
System.out.println(y); // Output: 7

y = 9;

System.out.println(x); // Output: 7
System.out.println(y); // Output: 9

// REFERENCE
int[] x = {2, 3, 5, 7};
int[] y = x;

for(int e : x)
    System.out.print(e + " "); // Output: 2, 3, 5, 7
System.out.println();

for(int e : y)
    System.out.print(e + " "); // Output: 2, 3, 5, 7
System.out.println();

y[0] = 1;

for(int e : x)
    System.out.print(e + " "); // Output: 1, 3, 5, 7
System.out.println();

for(int e : y)
    System.out.print(e + " "); // Output: 1, 3, 5, 7
System.out.println();

```

Pass by value diterapkan pada argumen bertipe data primitif dan prosesnya hanya menyalin isi memori (yang telah dialokasikan untuk suatu variabel), dan kemudian menyampaikan salinan tersebut kepada method yang bersangkutan (isi memory merupakan data sesungguhnya yang akan dioperasikan), karena yang disampaikan hanya salinan dari isi memori, maka perubahan yang terjadi pada variabel akibat proses didalam method tidak mempengaruhi nilai variabel asalnya di dalam memori. Sedangkan, pass by reference diterapkan pada argumen bertipe data array atau objek dan isi memori pada variabel array atau objek merupakan penunjuk alamat memori yang mengandung data sesungguhnya yang akan dioperasikan.

6. Responsibility Driven Design. Semua fungsi dan method yang ada harus mencerminkan perilaku lengkap yang dimiliki kelas tersebut. Semua fungsi/method tersebut bertanggung jawab terhadap maintenance atribut yang dimiliki kelas.

III. GUIDED

Buatlah package baru dengan format com.nama.praktikumpbo.pertemuan3.guided, lalu ikuti langkah langkah di bawah:

DASAR ARRAY

Untuk membuat array kosong di Java, pertama kita tuliskan dulu tipe data yang nantinya akan ditampung array yang kita buat. Setelah itu, jangan lupa berikan kurung siku terbuka dan tertutup []. Setelah itu, tulis nama yang akan kita gunakan untuk mereferensikan array tersebut. Terakhir, kita harus menuliskan tipe data array tersebut lagi dan kali ini di dalam kurung siku kita tuliskan jumlah elemen (isi) maksimum yang dapat ditampung oleh array tersebut.

Buatlah sebuah package baru di dalam package guided dan beri nama array. Buat class baru di dalam package array tersebut dan beri nama ArrayBasic. Salin kode berikut:



```
package com.iqbal.praktikumpbo.pertemuan3.guided;

/**
 * Hello world!
 *
 */
public class ArrayBasic
{
    public static void main( String[] args )
    {
        //tipeData[] namaArray = new tipeData[panjangArray];
        int[] numbers = new int[10];

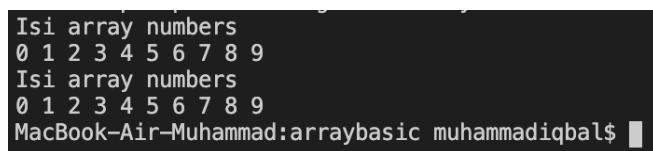
        //Array diisi dengan bilangan 0 sampai 9
        for(int i=0; i<10; i++)
            numbers[i] = i;

        //output seluruh isi array (cara 1)
        System.out.println("Isi array numbers");
        for(int i=0; i<numbers.length; i++)
            System.out.print(numbers[i]+ " ");
        System.out.println();

        //output seluruh isi array (cara2)
        System.out.println("Isi array numbers");
        for(int number : numbers)
            System.out.print(number+ " ");
        System.out.println();
    }
}
```

Pada contoh di atas, array kita beri nama numbers. dan array numbers ini dapat menampung sejumlah 10 bilangan bulat. Untuk mencetak isi array, kita bisa memanfaatkan for loop. Ada 2 variasi for loop, anda bebas ingin menggunakan yang mana.

Hasil running kodennya:

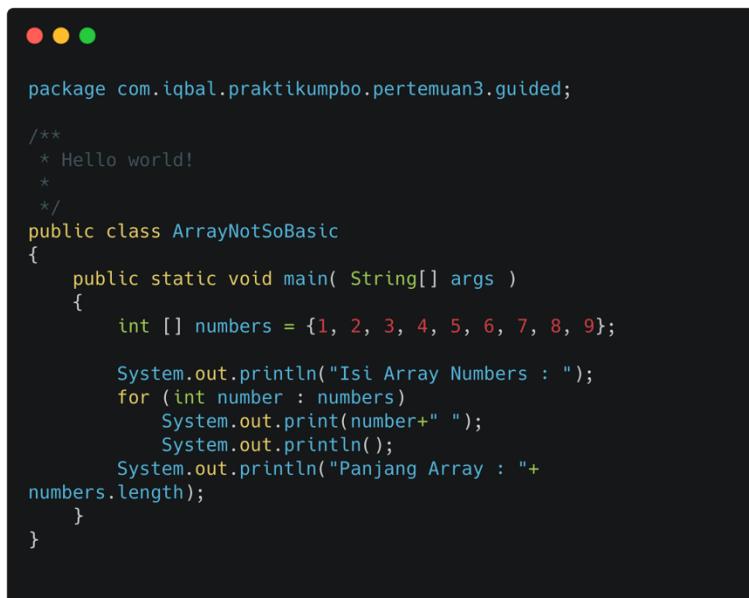


```
Isi array numbers
0 1 2 3 4 5 6 7 8 9
Isi array numbers
0 1 2 3 4 5 6 7 8 9
MacBook-Air-Muhammad:arraybasic muhammadiqbal$
```

ARRAY DENGAN NILAI YANG SUDAH DITENTUKAN DARI AWAL

Untuk membuat sebuah array yang sudah langsung memiliki nilai sejak ia dibuat, pertama tuliskan tipe datanya terlebih dahulu diikuti dengan 2 kurung siku []. Setelah itu jangan lupa tulis nama array-nya. Lalu, tulis = {}. Nah, untuk memasukan sesuatu ke array ini, anda tinggal mengetikan saja apa yang anda masukan ke array tersebut ke dalam {} tadi. Gunakan tanda koma (,) sebagai pemisah antara elemen/nilai index satu dengan index lainnya.

Di package yang sama, buatlah class baru lagi dan beri nama ArrayNotSoBasic, lalu salin kode berikut:

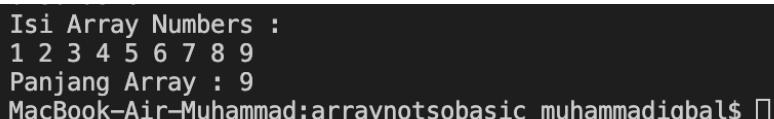


```
package com.iqbaliqbal.praktikumpbo.pertemuan3.guided;

/**
 * Hello world!
 *
 */
public class ArrayNotSoBasic
{
    public static void main( String[] args )
    {
        int [] numbers = {1, 2, 3, 4, 5, 6, 7, 8, 9};

        System.out.println("Isi Array Numbers : ");
        for (int number : numbers)
            System.out.print(number+" ");
        System.out.println();
        System.out.println("Panjang Array : "+
numbers.length);
    }
}
```

Pada contoh di atas, kita membuat array dengan nama numbers yang menampung bilangan bulat 1, 2, 3, 4, 5, 6, 7, 8. Jika anda membuat array dengan cara seperti ini, anda tidak perlu menuliskan berapa panjang array-nya, karena panjang array akan menyesuaikan, tergantung anda memasukan berapa nilai atau object ke dalam array tersebut saat anda membuatnya. Untuk mengecek panjang array, anda bisa menuliskan nama array tersebut, lalu menambahkan .length setelahnya. Pada contoh di atas, karena nama array-nya adalah numbers, jadi kita menuliskan numbers.length



```
Isi Array Numbers :
1 2 3 4 5 6 7 8 9
Panjang Array : 9
MacBook-Air-Muhammad:arraynotsobasic muhammadiqbals ~
```

ARRAY KOSONG DENGAN PANJANG BERVARIASI

Selain 2 cara di atas, ada cara lain lagi yang bisa anda gunakan untuk membuat array. Pada cara ini, panjang array akan ditentukan oleh variable, yang mana berarti kita bisa membuat program yang memiliki array dengan panjang yang berbeda tergantung inputan user!

Masih di package yang sama, buatlah class baru dan beri nama ArrayVariableLength, lalu salin kode berikut:

```
● ● ●

package com.iqbal.praktikumpbo.pertemuan3.guided;
import java.util.Scanner;
/**
 * Hello world!
 *
 */
public class ArrayVariableLength
{
    public static void main( String[] args )
    {
        Scanner input = new Scanner(System.in);
        System.out.print("Masukkan Panjang Array: ");
        int lenght = input.nextInt();

        int[] numbers = new int[lenght];

        for(int i=0; i<lenght; i++)
            numbers[i] = i;

        System.out.println("Isi array numbers: ");
        for(int number : numbers)
            System.out.print(number + " ");
    }
}
```

Prinsipnya sama seperti metode pembuatan array pertama, hanya saja kita memasukan sebuah variable ke dalam [], bukan angka.

Hasil running kodanya:

```
ArrayVariableLength
Masukkan Panjang Array: 14
Isi array numbers:
0 1 2 3 4 5 6 7 8 9 10 11 12 13 MacBook-Air-Muhammad:ArrayList
SoBasic muhammadiqbal$
```

Sekarang, kita akan mencoba merapikan kode **ArrayVariableLength** ini. Menggunakan prinsip Enkapsulasi yang kita pelajari di awal pertemuan ini, jika ada serangkaian kode yang memiliki konteks atau tujuan yang sama, alangkah baiknya jika kita buat rangkaian kode tersebut menjadi sebuah fungsi. Mari kita coba terapkan prinsip tersebut! Source **kode ArrayVariableLength** setelah dirapikan:

```
● ● ●

package com.iqbaliqbal.praktikumpbo.pertemuan3.guided;

import java.util.Scanner;

public class ArrayVariableLength {
    public static void isiarray(int[] array){
        for (int i=0;i<array.length;i++)
            array[i]=i;
    }

    public static void lihatisiarray(int[] array){
        System.out.println("Isi array numbers : ");
        for (int number : array)
            System.out.print(number + " ");
    }

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

        System.out.print("Masukkan panjang array : ");
        int length = input.nextInt();

        int [] numbers = new int[length];

        isiarray(numbers);
        lihatisiarray(numbers);
    }
}
```

PENGGUNAAN ARRAY DALAM PBO

Buatlah package baru di luar package array tetapi masih di dalam package guided. Namai package tersebut restaurant. Anda berada di jalan yang benar jika package anda terlihat seperti ini: `com.nama.praktikumpbo.pertemuan3.guided.restaurant` Di dalam package tersebut, buat class baru bernama **Item**, lalu salin kode berikut:

```
● ● ●

package com.iqbal.praktikumpbo.pertemuan3.guided.restaurant;

public class Item {
    private String nama;
    private int harga;
    private String jenis;

    public Item(String nama, int harga, String jenis) {
        this.nama = nama;
        this.harga = harga;
        this.jenis = jenis;
    }

    public String getNama() {
        return nama;
    }

    public int getHarga() {
        return harga;
    }

    public String getJenis() {
        return jenis;
    }

    public String showDetails(){
        return getNama() + "(" + getJenis() + ") - Rp " +
getHarga();
    }
}
```

Masih di package yang sama, buat class baru dan beri nama Pembeli. Salin kode berikut:

```
package com.iqbal.praktikumpbo.pertemuan3.guided.restaurant;

public class Pembeli {
    private String nama;
    private Item[] historyPemesanan = new Item[99];
    private int itemsOrdered = 0;
    private int bayar = 0;

    public Pembeli(String nama) {
        this.nama = nama;
    }

    public String getNama() {
        return nama;
    }

    public void lakukanPemesanan(Item item){
        historyPemesanan[itemsOrdered]= item;
        itemsOrdered++;
        System.out.println(item.getNama()
        + " Berhasil Ditambahkan ke History Pemesanan Sdr " +
this.getNama());
    }

    public void lihatHistoryPemesanan(){
        if (itemsOrdered > 0) {
            System.out.println("History Pemesanan Sdr " +
this.getNama() + ":");

            for (int i = 0; i < itemsOrdered; i++) {
                Item pesanan = historyPemesanan[i];
                System.out.println("- " + pesanan.getNama() +
" " + pesanan.getJenis() + " ) Rp"
                + pesanan.getHarga());
            }
            System.out.println("");
        }else{
            System.out.println("Anda belum pernah memesan");
            System.out.println();
        }
    }

    public void kosongkanHistory(){
        System.out.println("Mereset History");
        this.historyPemesanan = new Item[99];
        this.itemsOrdered = 0;
        System.out.println("History Pemesanan Berhasil
Dikosongkan");
        System.out.println("");
    }

    public void totalHarga(int i){
        if(itemsOrdered > 0){
            System.out.println("Total Harga yang Harus
Dibayar" + this.getNama() + "=");
            //for (int i = 0; i < itemsOrdered; i++){}
            Item pesanan = historyPemesanan[i];
            System.out.println("- " + pesanan.getJenis()
            + "- Rp. " + pesanan.getHarga());
            bayar += pesanan.getHarga();
        }
        System.out.println("Total yang Harus Anda Bayar : Rp.
" + bayar);
        System.out.println(" ");
    }

    public void totalHarga() {
    }
}
```

Terakhir, buat class baru lagi bernama Main masih di package yang sama, lalu salin kode berikut:



```
package com.iqbal.praktikumpbo.pertemuan3.guided;

import java.util.Scanner;

public class Main {
    private static void tampilanMenu() {
        int nomorItem = 1;
        for (Item item : menu) {
            System.out.println(nomorItem + ". " +
item.showDetails());
            nomorItem++;
        }
        System.out.println();
    }

    public static void main(String[] args) {
        Scanner input = new Scanner(System.in);
        boolean keepLooping = true;

        System.out.println("Maukan Nama :");
        String namaPembeli = input.nextLine();
        Pembeli pembeli = new Pembeli(namaPembeli);

        System.out.println();
        do {
            System.out.println("==>Menu Utama<==");
            System.out.println("1) Pesan makanan");
            System.out.println("2) LIHAT History Pemesanan");
            System.out.println("3) Hapus History pemesanan");
            System.out.println("4) Total Pembayaran");
            System.out.println("5) Keluar");
            System.out.print("> ");

            int nav = input.nextInt();
            System.out.println();

            switch (nav) {
                case 1: // pesan makanan
                    tampilanMenu();
                    System.out.print("Pilih makanan yang ingin
dipesan: ");
                    int indexItem = input.nextInt() - 1;

                    // jika user menginput aneh aneh
                    if (indexItem < 0 || indexItem >= menu.length)
                        System.out.println("Error: Mohon masukkan
yang benar!");

                    // jika user benar
                    else
                        pembeli.lakukanPemesanan(menu[indexItem]);

                    System.out.println();
                    break;

                case 2: // Lihat history
                    pembeli.lihatHistoryPemesanan();
                    break;

                case 3: // kosongkan history
                    pembeli.kosongkanHistory();
                    break;

                case 4://total harga yang harus dibayar
                    System.out.println("Total Harga yang Harus
Anda Bayar Sebesar : ");
                    pembeli.totalHarga();
                    break;

                case 5: // keluar
                    System.out.println("Terimakasih sudah mampir!
^__^");
                    keepLooping = false;
                    break;

                default: // jika user menginput aneh aneh
                    System.out.println("Tolong hanya masukkan
angka 1-4 saja!");
                    System.out.println();
                    break;
            }
        } while (keepLooping);
    }

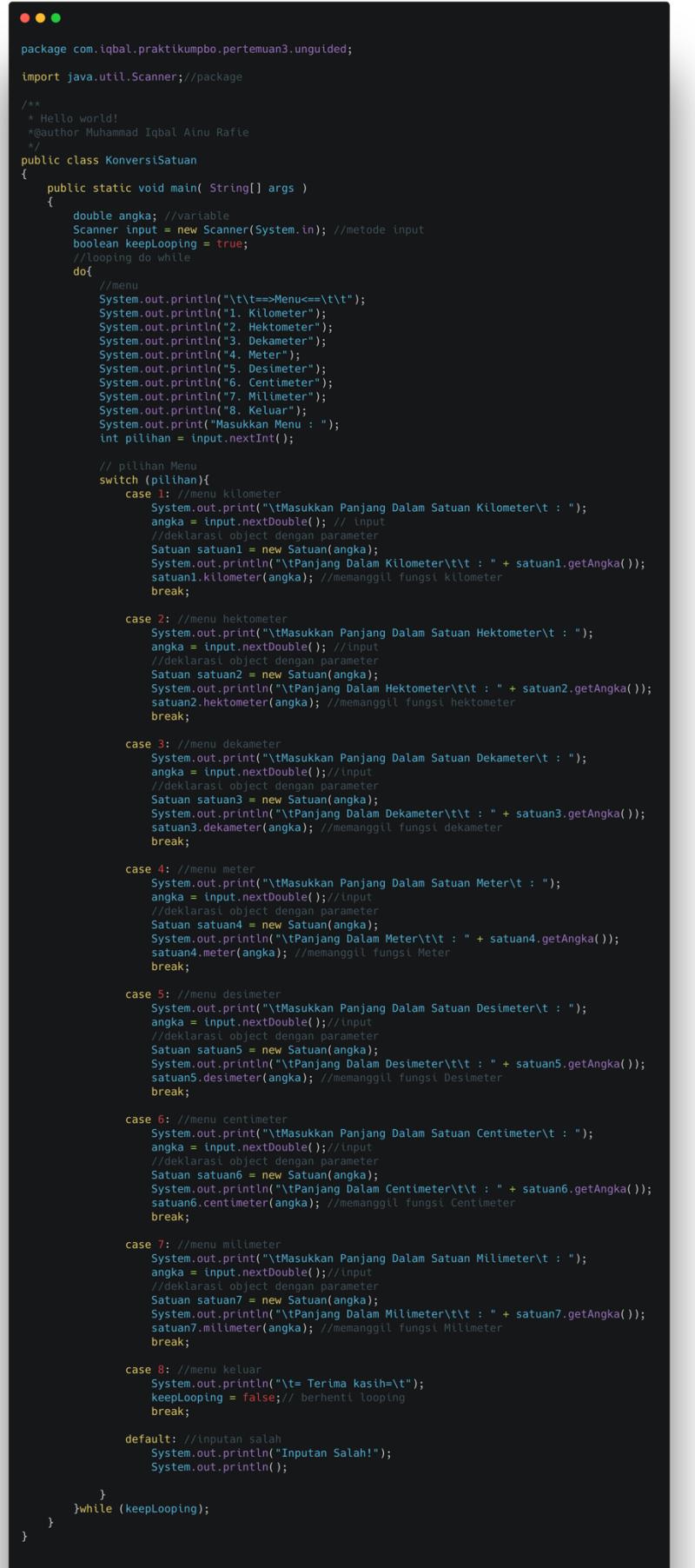
    private static Item[] menu = {
        new Item("Mendoan", 1_000, "Makanan"),
        new Item("Aneka Sayur", 4_500, "Makanan"),
        new Item("Ayam Goreng", 6_000, "Makanan"),
        new Item("Sate Ayam", 7_000, "Makanan"),
        new Item("Es teh hangat", 3_000, "Minuman"), };
}
```

IV. UNGUIDED

Seperti biasa, buat package baru di dalam package pertemuan3 dan beri nama unguided. Anda berada di jalan yang benar jika package anda terlihat seperti ini: com.nama.praktikumpbo.pertemuan3.unguided ;)

1. Buatlah sebuah program sederhana dengan tema bebas yang melibatkan pembuatan class dan object! Syarat dan ketentuan:
 - Minimal 2 file, satu untuk class object dan satu untuk class main.
 - Sertakan source code dan screenshot output saat running program di laporan praktikum.
 - Jangan lupa beri penjelasan tentang alur kerja program di laporan.
 - Nilai plus jika program memiliki menu.
 - Nilai plus plus jika source code rapi, enak dilihat, dan mudah dipahami saat pertama kali dibaca.

- Class KonversiSatuan



```
package com.iqbal.praktikumpto.pertemuan3.unguided;

import java.util.Scanner;//package

/**
 * Hello world!
 * @author Muhammad Iqbal Ainu Rafie
 */
public class KonversiSatuan
{
    public static void main( String[] args )
    {
        double angka; //variable
        Scanner input = new Scanner(System.in); //metode input
        boolean keepLooping = true;
        //Looping do while
        do{
            //menu
            System.out.println("\t\t==>Menu<==\t\t");
            System.out.println("1. Kilometer");
            System.out.println("2. Hektometer");
            System.out.println("3. Dekameter");
            System.out.println("4. Meter");
            System.out.println("5. Desimeter");
            System.out.println("6. Centimeter");
            System.out.println("7. Millimeter");
            System.out.println("8. Keluar");
            System.out.print("Masukkan Menu : ");
            int pilihan = input.nextInt();

            // pilihan Menu
            switch (pilihan){
                case 1: //menu kilometer
                    System.out.print("\tMasukkan Panjang Dalam Satuan Kilometer\t : ");
                    angka = input.nextDouble(); // input
                    //deklarasi object dengan parameter
                    Satuan satuan1 = new Satuan(angka);
                    System.out.println("\tPanjang Dalam Kilometer\t\t : " + satuan1.getAngka());
                    satuan1.kilometer(angka); //memanggil fungsi kilometer
                    break;

                case 2: //menu hektometer
                    System.out.print("\tMasukkan Panjang Dalam Satuan Hektometer\t : ");
                    angka = input.nextDouble(); // input
                    //deklarasi object dengan parameter
                    Satuan satuan2 = new Satuan(angka);
                    System.out.println("\tPanjang Dalam Hektometer\t\t : " + satuan2.getAngka());
                    satuan2.hektometer(angka); //memanggil fungsi hektometer
                    break;

                case 3: //menu dekameter
                    System.out.print("\tMasukkan Panjang Dalam Satuan Dekameter\t : ");
                    angka = input.nextDouble(); // input
                    //deklarasi object dengan parameter
                    Satuan satuan3 = new Satuan(angka);
                    System.out.println("\tPanjang Dalam Dekameter\t\t : " + satuan3.getAngka());
                    satuan3.dekameter(angka); //memanggil fungsi dekameter
                    break;

                case 4: //menu meter
                    System.out.print("\tMasukkan Panjang Dalam Satuan Meter\t : ");
                    angka = input.nextDouble(); // input
                    //deklarasi object dengan parameter
                    Satuan satuan4 = new Satuan(angka);
                    System.out.println("\tPanjang Dalam Meter\t\t : " + satuan4.getAngka());
                    satuan4.meter(angka); //memanggil fungsi Meter
                    break;

                case 5: //menu desimeter
                    System.out.print("\tMasukkan Panjang Dalam Satuan Desimeter\t : ");
                    angka = input.nextDouble(); // input
                    //deklarasi object dengan parameter
                    Satuan satuan5 = new Satuan(angka);
                    System.out.println("\tPanjang Dalam Desimeter\t\t : " + satuan5.getAngka());
                    satuan5.desimeter(angka); //memanggil fungsi Desimeter
                    break;

                case 6: //menu centimeter
                    System.out.print("\tMasukkan Panjang Dalam Satuan Centimeter\t : ");
                    angka = input.nextDouble(); // input
                    //deklarasi object dengan parameter
                    Satuan satuan6 = new Satuan(angka);
                    System.out.println("\tPanjang Dalam Centimeter\t\t : " + satuan6.getAngka());
                    satuan6.centimeter(angka); //memanggil fungsi Centimeter
                    break;

                case 7: //menu milimeter
                    System.out.print("\tMasukkan Panjang Dalam Satuan Milimeter\t : ");
                    angka = input.nextDouble(); // input
                    //deklarasi object dengan parameter
                    Satuan satuan7 = new Satuan(angka);
                    System.out.println("\tPanjang Dalam Milimeter\t\t : " + satuan7.getAngka());
                    satuan7.milimeter(angka); //memanggil fungsi Milimeter
                    break;

                case 8: //menu keluar
                    System.out.println("\t= Terima kasih=\t");
                    keepLooping = false;// berhenti looping
                    break;

                default: //inputan salah
                    System.out.println("Inputan Salah!");
                    System.out.println();
            }
        }while (keepLooping);
    }
}
```

Class Satuan

```
package com.lqbal.praktikumbo.pertemuan3.unguided;

public class Satuan {
    //variabel
    double kilometer = 0;
    double hektometer = 0;
    double dekometer = 0;
    double meter = 0;
    double desimeter = 0;
    double centimeter = 0;
    double millimeter = 0;
    double angka;

    //fungsi 1 densus parameter
    public Satuan(double angka){
        this.angka = angka;
    }
    //getter dan setter
    public double getAngka(){
        return angka;
    }
    //fungsi kilometer
    void satuan(double angka){
        hektometer = (angka/10);
        dekometer = (angka*100);
        meter = (angka*1000);
        desimeter = (angka*10000);
        centimeter = (angka*100000);
        millimeter = (angka*1000000);
        System.out.println("\tSatuan dalam hektometer\t : "
+hektometer+"hm");
        System.out.println("\tSatuan dalam dekometer\t : "
+dekometer+"dm");
        System.out.println("\tSatuan dalam meter\t : "+meter+
"m");
        System.out.println("\tSatuan dalam desimeter\t : "
+desimeter+"dm");
        System.out.println("\tSatuan dalam centimeter\t : "
+centimeter+"cm");
        System.out.println("\tSatuan dalam millimeter\t : "
+millimeter+"mm");
        System.out.println("");
    }
    //fungsi hektometer
    void hektometer(double angka){
        kilometer = (angka/10);
        desimeter = (angka*10000);
        meter = (angka*1000);
        desimeter = (angka*10000);
        centimeter = (angka*100000);
        millimeter = (angka*1000000);
        System.out.println("\tSatuan dalam kilometer\t : "
+kilometer+"km");
        System.out.println("\tSatuan dalam hektometer\t : "
+hektometer+"hm");
        System.out.println("\tSatuan dalam meter\t : "+meter+
"m");
        System.out.println("\tSatuan dalam desimeter\t : "
+desimeter+"dm");
        System.out.println("\tSatuan dalam centimeter\t : "
+centimeter+"cm");
        System.out.println("\tSatuan dalam millimeter\t : "
+millimeter+"mm");
        System.out.println("");
    }
    //fungsi dekometer
    void dekometer(double angka){
        kilometer = (angka/100);
        hektometer = (angka/10);
        meter = (angka*10);
        desimeter = (angka*1000);
        centimeter = (angka*10000);
        millimeter = (angka*100000);
        System.out.println("\tSatuan dalam kilometer\t : "
+kilometer+"km");
        System.out.println("\tSatuan dalam hektometer\t : "
+hektometer+"hm");
        System.out.println("\tSatuan dalam dekometer\t : "
+dekometer+"dm");
        System.out.println("\tSatuan dalam meter\t : "+meter+
"m");
        System.out.println("\tSatuan dalam desimeter\t : "
+desimeter+"dm");
        System.out.println("\tSatuan dalam centimeter\t : "
+centimeter+"cm");
        System.out.println("\tSatuan dalam millimeter\t : "
+millimeter+"mm");
        System.out.println("");
    }
    //fungsi meter
    void meter(double angka){
        kilometer = (angka/1000);
        hektometer = (angka/100);
        desimeter = (angka*100);
        meter = (angka*100);
        desimeter = (angka*100);
        centimeter = (angka*1000);
        millimeter = (angka*10000);
        System.out.println("\tSatuan dalam kilometer\t : "
+kilometer+"km");
        System.out.println("\tSatuan dalam hektometer\t : "
+hektometer+"hm");
        System.out.println("\tSatuan dalam dekometer\t : "
+dekometer+"dm");
        System.out.println("\tSatuan dalam meter\t : "+meter+
"m");
        System.out.println("\tSatuan dalam desimeter\t : "
+desimeter+"dm");
        System.out.println("\tSatuan dalam centimeter\t : "
+centimeter+"cm");
        System.out.println("\tSatuan dalam millimeter\t : "
+millimeter+"mm");
        System.out.println("");
    }
    //fungsi desimeter
    void desimeter(double angka){
        kilometer = (angka/10000);
        hektometer = (angka/1000);
        dekometer = (angka/100);
        meter = (angka/10);
        desimeter = (angka*10);
        centimeter = (angka*100);
        millimeter = (angka*1000);
        System.out.println("\tSatuan dalam kilometer\t : "
+kilometer+"km");
        System.out.println("\tSatuan dalam hektometer\t : "
+hektometer+"hm");
        System.out.println("\tSatuan dalam dekometer\t : "
+dekometer+"dm");
        System.out.println("\tSatuan dalam meter\t : "+meter+
"m");
        System.out.println("\tSatuan dalam desimeter\t : "
+desimeter+"dm");
        System.out.println("\tSatuan dalam centimeter\t : "
+centimeter+"cm");
        System.out.println("\tSatuan dalam millimeter\t : "
+millimeter+"mm");
        System.out.println("");
    }
    //fungsi centimeter
    void centimeter(double angka){
        kilometer = (angka/100000);
        hektometer = (angka/10000);
        dekometer = (angka/1000);
        meter = (angka/100);
        desimeter = (angka/10);
        millimeter = (angka*10);
        System.out.println("\tSatuan dalam kilometer\t : "
+kilometer+"km");
        System.out.println("\tSatuan dalam hektometer\t : "
+hektometer+"hm");
        System.out.println("\tSatuan dalam dekometer\t : "
+dekometer+"dm");
        System.out.println("\tSatuan dalam meter\t : "+meter+
"m");
        System.out.println("\tSatuan dalam desimeter\t : "
+desimeter+"dm");
        System.out.println("\tSatuan dalam centimeter\t : "
+centimeter+"cm");
        System.out.println("\tSatuan dalam millimeter\t : "
+millimeter+"mm");
        System.out.println("");
    }
    //fungsi millimeter
    void millimeter(double angka){
        kilometer = (angka/1000000);
        hektometer = (angka/100000);
        dekometer = (angka/10000);
        meter = (angka/1000);
        desimeter = (angka/100);
        centimeter = (angka/10);
        System.out.println("\tSatuan dalam kilometer\t : "
+kilometer+"km");
        System.out.println("\tSatuan dalam hektometer\t : "
+hektometer+"hm");
        System.out.println("\tSatuan dalam dekometer\t : "
+dekometer+"dm");
        System.out.println("\tSatuan dalam meter\t : "+meter+
"m");
        System.out.println("\tSatuan dalam desimeter\t : "
+desimeter+"dm");
        System.out.println("\tSatuan dalam centimeter\t : "
+centimeter+"cm");
        System.out.println("");
    }
}
```

Output menu 1 (Kilometer)

The screenshot shows a Java development environment with two open files: KonversiSatuan.java and Satuan.java. The terminal window displays the execution of the program, which asks for input and prints conversion results for kilometers.

```
src/main/java/com/iqbal/praktikumpbo/pertemuan3/unguided> KonversiSatuan.java > {} com.iqbal.praktikumpbo.pertemuan3.unguided
1. Kilometer
2. Hektometer
3. Dekameter
4. Meter
5. Desimeter
6. Centimeter
7. Milimeter
8. Keluar
Masukkan Menu : 1
Masukkan Panjang Dalam Satuan Kilometer : 120
Panjang Dalam Kilometer : 120.0
Satuan dalam hektometer : 12000.0hm
Satuan dalam dekameter : 12000.0dam
Satuan dalam meter : 120000.0m
Satuan dalam desimeter : 1200000.0dm
Satuan dalam centimeter : 12000000.0cm
Satuan dalam milimeter : 120000000.0mm
1. Kilometer
2. Hektometer
3. Dekameter
4. Meter
5. Desimeter
6. Centimeter
7. Milimeter
8. Keluar
```

Output menu 2 (Hektometer)

The screenshot shows a Java development environment with two open files: KonversiSatuan.java and Satuan.java. The terminal window displays the execution of the program, which asks for input and prints conversion results for hectometers.

```
src/main/java/com/iqbal/praktikumpbo/pertemuan3/unguided> KonversiSatuan.java > {} com.iqbal.praktikumpbo.pertemuan3.unguided
1. Kilometer
2. Hektometer
3. Dekameter
4. Meter
5. Desimeter
6. Centimeter
7. Milimeter
8. Keluar
Masukkan Menu : 2
Masukkan Panjang Dalam Satuan Hektometer : 120
Panjang Dalam Hektometer : 120.0
Satuan dalam kilometer : 12.0km
Satuan dalam dekameter : 1200.0dam
Satuan dalam meter : 12000.0m
Satuan dalam desimeter : 120000.0dm
Satuan dalam centimeter : 1200000.0cm
Satuan dalam milimeter : 12000000.0mm
1. Kilometer
2. Hektometer
3. Dekameter
4. Meter
5. Desimeter
6. Centimeter
7. Milimeter
```

Output menu 3 (Dekameter)

```
src > main > java > com.iqbal.praktikumpbo.pertemuan3 > unguided > KonversiSatuan.java > () com.iqbal.praktikumpbo.pertemuan3.unguided
```

```
int pilihan = input.nextInt();

// pilihan Menu
switch (pilihan){
    case 1: //menu kilometer
        System.out.print("\nMasukkan Panjang Dalam Satuan Kilometer\t : ");
        angka = input.nextDouble(); // input
        //deklarasi object dengan parameter
        Satuan satuan1 = new Satuan(angka);
        System.out.println("\nPanjang Dalam Kilometer\t : " + satuan1.getAngka());
        satuan1.kilometer(angka); //memanggil fungsi kilometer
        break;

    case 2: //menu hektometer
        System.out.print("\nMasukkan Panjang Dalam Satuan Hektometer\t : ");
        angka = input.nextDouble(); //input
        //deklarasi object dengan parameter
        Satuan satuan2 = new Satuan(angka);
        System.out.println("\nPanjang Dalam Hektometer\t : " + satuan2.getAngka());
        satuan2.hektometer(angka); //memanggil fungsi hektometer
        break;
}
```

==>Menu<==

1. Kilometer
2. Hektometer
3. Dekameter
4. Meter
5. Desimeter
6. Centimeter
7. Milimeter
8. Keluar

Masukkan Menu : 3

```
Masukkan Panjang Dalam Satuan Dekameter : 120
Panjang Dalam Dekameter : 120.0
Satuan dalam Kilometer : 1.2km
Satuan dalam hektometer : 12.0hm
Satuan dalam meter : 120.0m
Satuan dalam desimeter : 12000.0dm
Satuan dalam centimeter : 1200000.0cm
Satuan dalam milimeter : 12000000.0mm
```

==>Menu<==

1. Kilometer
2. Hektometer
3. Dekameter
4. Meter
5. Desimeter
6. Centimeter
7. Milimeter
8. Keluar

Output menu 4 (Meter)

```
src > main > java > com.iqbal.praktikumpbo.pertemuan3 > unguided > KonversiSatuan.java > () com.iqbal.praktikumpbo.pertemuan3.unguided
```

```
int pilihan = input.nextInt();

// pilihan Menu
switch (pilihan){
    case 1: //menu kilometer
        System.out.print("\nMasukkan Panjang Dalam Satuan Kilometer\t : ");
        angka = input.nextDouble(); // input
        //deklarasi object dengan parameter
        Satuan satuan1 = new Satuan(angka);
        System.out.println("\nPanjang Dalam Kilometer\t : " + satuan1.getAngka());
        satuan1.kilometer(angka); //memanggil fungsi kilometer
        break;

    case 2: //menu hektometer
        System.out.print("\nMasukkan Panjang Dalam Satuan Hektometer\t : ");
        angka = input.nextDouble(); //input
        //deklarasi object dengan parameter
        Satuan satuan2 = new Satuan(angka);
        System.out.println("\nPanjang Dalam Hektometer\t : " + satuan2.getAngka());
        satuan2.hektometer(angka); //memanggil fungsi hektometer
        break;
}
```

==>Menu<==

1. Kilometer
2. Hektometer
3. Dekameter
4. Meter
5. Desimeter
6. Centimeter
7. Milimeter
8. Keluar

Masukkan Menu : 4

```
Masukkan Panjang Dalam Satuan Meter : 120
Panjang Dalam Meter : 120.0
Satuan dalam Kilometer : 0.12km
Satuan dalam hektometer : 1.2hm
Satuan dalam dekameter : 12.0dam
Satuan dalam desimeter : 1200.0dm
Satuan dalam centimeter : 120000.0cm
Satuan dalam milimeter : 1200000.0mm
```

==>Menu<==

1. Kilometer
2. Hektometer
3. Dekameter
4. Meter
5. Desimeter
6. Centimeter
7. Milimeter
8. Keluar

Output menu 5 (Desimeter)

```
src > main > java > com > iqbal > praktikumpbo > pertemuan3 > unguided > KonversiSatuan.java > () com.iqbal.praktikumpbo.pertemuan3.unguided
```

```
int pilihan = input.nextInt();

// pilihan Menu
switch (pilihan){
    case 1: //menu kilometer
        System.out.print("\nMasukkan Panjang Dalam Satuan Kilometer\t : ");
        angka = input.nextDouble(); // input
        //deklarasi object dengan parameter
        Satuan satuan1 = new Satuan(angka);
        System.out.println("\nPanjang Dalam Kilometer\t : " + satuan1.getAngka());
        satuan1.kilometer(angka); //memanggil fungsi kilometer
        break;

    case 2: //menu hektometer
        System.out.print("\nMasukkan Panjang Dalam Satuan Hektometer\t : ");
        angka = input.nextDouble(); //input
        //deklarasi object dengan parameter
        Satuan satuan2 = new Satuan(angka);
        System.out.println("\nPanjang Dalam Hektometer\t : " + satuan2.getAngka());
        satuan2.hektometer(angka); //memanggil fungsi hektometer
        break;
}
```

TERMINAL PROBLEMS ③ OUTPUT DEBUG CONSOLE

```
==>Menu<==  
1. Kilometer  
2. Hektometer  
3. Dekameter  
4. Meter  
5. Desimeter  
6. Centimeter  
7. Milimeter  
8. Keluar  
Masukkan Menu : 5  
Masukkan Panjang Dalam Satuan Desimeter : 120  
Panjang Dalam Desimeter : 120.0  
Satuan dalam kilometer : 0.012km  
Satuan dalam hektometer : 0.12hm  
Satuan dalam dekameter : 1.2dam  
Satuan dalam meter : 1.2m  
Satuan dalam centimeter : 1200.0cm  
Satuan dalam milimeter : 12000.0mm  
==>Menu<==  
1. Kilometer  
2. Hektometer  
3. Dekameter  
4. Meter  
5. Desimeter  
6. Centimeter  
7. Milimeter  
8. Keluar
```

Ln 1, Col 1 Spaces: 4 UTF-8 LF Java ⚡ JavaSE-1.7

Output menu 6 (Centimeter)

```
src > main > java > com > iqbal > praktikumpbo > pertemuan3 > unguided > KonversiSatuan.java > () com.iqbal.praktikumpbo.pertemuan3.unguided
```

```
int pilihan = input.nextInt();

// pilihan Menu
switch (pilihan){
    case 1: //menu kilometer
        System.out.print("\nMasukkan Panjang Dalam Satuan Kilometer\t : ");
        angka = input.nextDouble(); // input
        //deklarasi object dengan parameter
        Satuan satuan1 = new Satuan(angka);
        System.out.println("\nPanjang Dalam Kilometer\t : " + satuan1.getAngka());
        satuan1.kilometer(angka); //memanggil fungsi kilometer
        break;

    case 2: //menu hektometer
        System.out.print("\nMasukkan Panjang Dalam Satuan Hektometer\t : ");
        angka = input.nextDouble(); //input
        //deklarasi object dengan parameter
        Satuan satuan2 = new Satuan(angka);
        System.out.println("\nPanjang Dalam Hektometer\t : " + satuan2.getAngka());
        satuan2.hektometer(angka); //memanggil fungsi hektometer
        break;
}
```

TERMINAL PROBLEMS ③ OUTPUT DEBUG CONSOLE

```
==>Menu<==  
1. Kilometer  
2. Hektometer  
3. Dekameter  
4. Meter  
5. Desimeter  
6. Centimeter  
7. Milimeter  
8. Keluar  
Masukkan Menu : 6  
Masukkan Panjang Dalam Satuan Centimeter : 120  
Panjang Dalam Centimeter : 120.0  
Satuan dalam kilometer : 0.0012km  
Satuan dalam hektometer : 0.012hm  
Satuan dalam dekameter : 0.12dam  
Satuan dalam meter : 1.2m  
Satuan dalam desimeter : 12.0dm  
Satuan dalam milimeter : 1200.0mm  
==>Menu<==  
1. Kilometer  
2. Hektometer  
3. Dekameter  
4. Meter  
5. Desimeter  
6. Centimeter  
7. Milimeter  
8. Keluar
```

Ln 1, Col 1 Spaces: 4 UTF-8 LF Java ⚡ JavaSE-1.7

Output menu 7 (Milimeter)

The screenshot shows a Java development environment with two files open: KonversiSatuan.java and Satuan.java. The KonversiSatuan.java file contains a main method that prints a menu and calls a method from Satuan.java. The Satuan.java file defines a class with methods for converting between Kilometer, Hektometer, Dekameter, Meter, Desimeter, Centimeter, and Milimeter. The output terminal shows the execution of the program, displaying the menu and the conversion of 120 units from Milimeter to other units.

```
src > main > java > com > iqbal > praktikumpbo > pertemuan3 > unguided > KonversiSatuan.java > () com.iqbal.praktikumpbo.pertemuan3.unguided
```

```
int pilihan = input.nextInt();

// pilihan Menu
switch (pilihan){
    case 1: //menu kilometer
        System.out.print("\nMasukkan Panjang Dalam Satuan Kilometer\t : ");
        angka = input.nextDouble(); // input
        //deklarasi object dengan parameter
        Satuan satuan1 = new Satuan(angka);
        System.out.println("\nPanjang Dalam Kilometer\t : " + satuan1.getAngka());
        satuan1.kilometer(angka); //memanggil fungsi kilometer
        break;

    case 2: //menu hektometer
        System.out.print("\nMasukkan Panjang Dalam Satuan Hektometer\t : ");
        angka = input.nextDouble(); //input
        //deklarasi object dengan parameter
        Satuan satuan2 = new Satuan(angka);
        System.out.println("\nPanjang Dalam Hektometer\t : " + satuan2.getAngka());
        satuan2.hektometer(angka); //memanggil fungsi hektometer
        break;
}
```

==>Menu<==

```
1. Kilometer
2. Hektometer
3. Dekameter
4. Meter
5. Desimeter
6. Centimeter
7. Milimeter
8. Keluar
Masukkan Menu : 7
Masukkan Panjang Dalam Satuan Milimeter : 120
Panjang Dalam Milimeter : 120.0
Satuan dalam Kilometer : 1.2E-3km
Satuan dalam hektometer : 0.0012hm
Satuan dalam dekameter : 0.012dam
Satuan dalam meter : 0.12m
Satuan dalam desimeter : 1.2dm
Satuan dalam centimeter : 12.0cm
```

==>Menu<==

```
1. Kilometer
2. Hektometer
3. Dekameter
4. Meter
5. Desimeter
6. Centimeter
7. Milimeter
8. Keluar
```

Ln 1, Col 1 Spaces: 4 UTF-8 LF Java JavaSE-1.7

Output menu 8 (keluar)

The screenshot shows a terminal window with a black background. It displays the final menu options and a message of thanks after exiting the program. The prompt at the bottom indicates the session is running on a MacBook Air.

```
==>Menu<==

1. Kilometer
2. Hektometer
3. Dekameter
4. Meter
5. Desimeter
6. Centimeter
7. Milimeter
8. Keluar
Masukkan Menu : 8
= Terima kasih=
MacBook-Air-Muhammad:KONVERSI SATUAN muhammadiqbal$
```