

LAPORAN PRAKTIKUM
ALGORITMA DAN PEMROGRAMAN LANJUT
PERTEMUAN KE-8



Disusun Oleh :

NAMA : Raden Isnawan Argi Aryasatya
NIM : 195410257
JURUSAN : Informatika
JENJANG : S1

SEMESTER ANTARA
Sekolah Tinggi Manajemen Informatika Komputer
AKAKOM
YOGYAKARTA

2021

PERTEMUAN KE-8 (METHOD)

TUJUAN

Mahasiswa dapat memahami, membuat dan menyelesaikan kasus dengan menggunakan method tanpa parameter

DASAR TEORI

Fungsi digunakan untuk mempermudah didalam membuat sebuah program, terutama program yang besar dan banyak melakukan beberapa hal yang sama. Fungsi memiliki ciri-ciri sebagai berikut:

1. Memiliki nama dari fungsi tersebut
2. Memiliki tugas spesifik tertentu
3. Memiliki sekumpulan statement atau perintah untuk melakukan tugas tersebut.
4. Mengembalikan sebuah nilai kepada fungsi lain yang memanggil atau menggunakannya (jika perlu).

Dalam bahasa java terdapat 2 macam fungsi yaitu :

1. Fungsi yang mengembalikan/menghasilkan nilai (non void function)
2. Fungsi tidak mengembalikan/menghasilkan nilai (void function)

PRAKTIK 1

```
Method1.java x
public class Method1{
    public static void cetak(){
        System.out.println("STMIK AKAKOM");
    }
    public static void main(String args[]){
        cetak();
    }
}
```

output:

```
C:\Windows\system32\cmd.exe
STMIK AKAKOM
Press any key to continue . . . _
```

penjelasan:

program di atas adalah contoh penerapan method yang sangat sederhana. Pertama, kita deklarasikan dulu nama class nya dengan menuliskan baris kode **public class Method1**. Kemudian langsung kita buat methodnya. Method yang kita buat di program ini adalah jenis method void yaitu method yang tidak menghasilkan nilai. Method itu kita tulis dengan kode **public static void cetak()** yang dimana method itu nantinya akan menampilkan tulisan atau sebuah pesan yaitu "STMIK AKAKOM" dengan baris kode **System.out.println("STMIK AKAKOM")**. Lalu kita buat main method dengan mengawali eksekusi dengan **public static void main(String args[])**. Di dalam main program tersebut, kita panggil method tersebut dengan kode **cetak()** supaya tulisan "STMIK AKAKOM" dapat ditampilkan di program.

PRAKTIK 2 DI HALAMAN SELANJUTNYA

PRAKTIK 2

```
Method2.java x Method1.java
public class Method2{
    public static void cetakKalimat(){
        System.out.println("Di dalam method kalimat");
    }
    public static void main(String args[]){
        cetakKalimat();
        System.out.println("Di dalam main");
        cetakKalimat();
    }
}
```

output:

```
C:\Windows\system32\cmd.exe
Di dalam method kalimat
Di dalam main
Di dalam method kalimat
Press any key to continue . . .
```

penjelasan:

program di praktik 2 ini konsepnya sama dengan program di praktik 1 tadi. Perbedaannya di program praktik 2 ini kita memanggil method sebanyak dua kali (lebih dari sekali). Untuk membuat program tersebut, kita awali dengan deklarasi nama class yaitu dengan menuliskan baris kode **public class Method2**. Setelah itu, langsung kita buat method voidnya yaitu dengan menuliskan baris kode **public static void cetakKalimat()** yang di dalamnya memiliki sebuah value berupa tulisan "Di dalam method kalimat" yaitu dengan menuliskan **System.out.println("Di dalam method kalimat")**. Kemudian kita buat main method yang diawali dengan **public static void main(String args[])**. Terakhir, kita panggil method **cetakKalimat()**, kemudian sebuah tulisan baru tanpa method yaitu **System.out.println("Di dalam main")**, setelah itu kita panggil method **cetakKalimat()** lagi.

PRAKTIK 3

```
Method3.java × Method2.java Method1.java
public class Method3{
    public static void cetak(){
        System.out.println("STMIK AKAKOM");
    }
    public static void main(String args[]){
        for (int i = 0; i < 10; i++)
            cetak();
    }
}
```

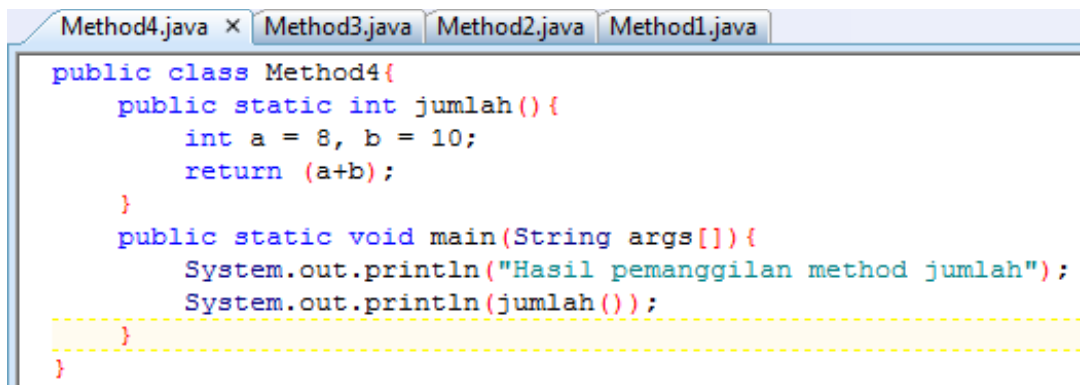
output:

[illegible]

penjelasan:

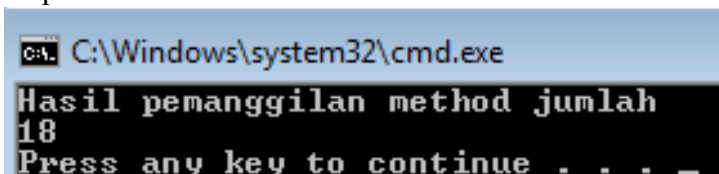
program di praktik 3 ini sedikit berbeda dengan program di praktik 1 dan praktik 2. Di program ini, kita menggunakan perulangan for. Perulangan for adalah perulangan yang digunakan ketika kita ingin mengeksekusi perintah program yang sama dengan jumlah proses perulangan yang sudah diketahui dengan mengacu pada kondisi yang sudah ditetapkan. Untuk membuat program di atas, bisa kita awali dengan mendeklarasikan nama class yaitu dengan menuliskan baris kode **public class Method3**. Kemudian langsung saja kita buat method bernama cetak() yaitu dengan menuliskan baris kode **public static void cetak()**. Method tersebut nantinya akan menampilkan kalimat "STMIK AKAKOM" yaitu dengan baris kode **System.out.println("STMIK AKAKOM")**. Program kita lanjutkan dengan membuat main method yang eksekusinya diawali dengan **public static void main(String args[])**. Lalu kita buat perulangannya yaitu **for (int i = 0; i < 10; i++)** yang fungsinya untuk mengulang pesan yang ada di dalam method cetak(). **i=1** berarti perulangan i dimulai dari 1. **i<=10** berarti selama i bernilai kurang dari atau sama dengan 10, maka perulangan akan terus berjalan. **i++** fungsinya adalah untuk menambah satu (+1) nilai hitungan pada setiap perulangan. Terakhir, kita panggil method cetak() sehingga muncul pesan berupa "STMIK AKAKOM" sebanyak 10 kali.

PRAKTIK 4



```
Method4.java x Method3.java Method2.java Method1.java
public class Method4{
    public static int jumlah(){
        int a = 8, b = 10;
        return (a+b);
    }
    public static void main(String args[]){
        System.out.println("Hasil pemanggilan method jumlah");
        System.out.println(jumlah());
    }
}
```

output:



```
C:\Windows\system32\cmd.exe
Hasil pemanggilan method jumlah
18
Press any key to continue . . .
```

penjelasan:

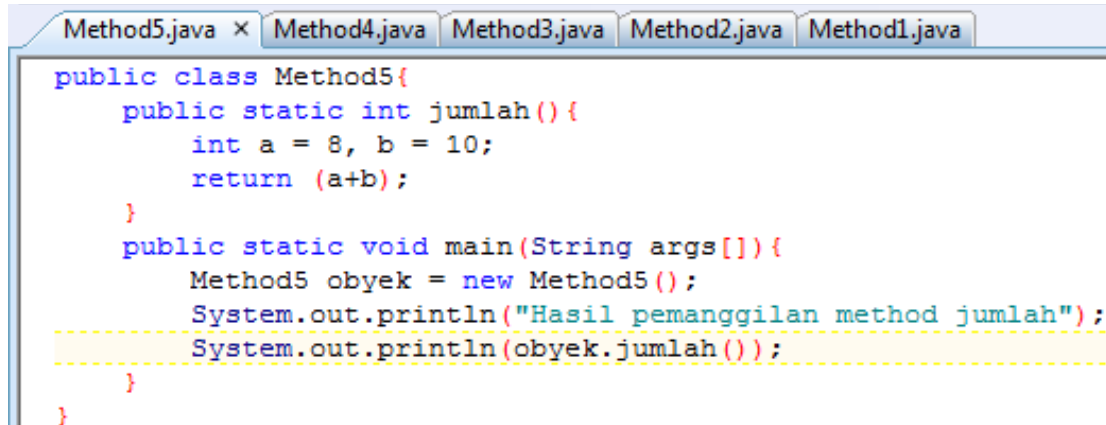
program di praktik 4 ini menggunakan jenis method yang berbeda dari method yang digunakan di praktik 1 sampai praktik 3. Di praktik 4 ini, kita menggunakan method non void yang membawa dan menghasilkan suatu nilai. Untuk membuat program tersebut, kita deklarasi dulu nama class dengan menuliskan **public class Method4**. Kemudian langsung saja kita buat method bernama jumlah dengan menuliskan baris kode **public static int jumlah()**. Di dalam method tersebut kita deklarasi dua variabel bertipe data integer yaitu variabel a dengan nilai 8 dan variabel b dengan nilai 10, kedua variabel itu kita tulis dengan baris kode **int a = 8, b = 10**. Setelah itu, kedua variabel tersebut kita kembalikan nilainya dengan return.

Method yang tidak menggunakan keyword void (method non void) harus mendeklarasikan return value dalam body atau block kode. Untuk dapat mendeklarasikan return value kita hanya perlu menuliskan sintaks **return returnValue;**. Di kasus program atas kita tulis **return (a+b)** untuk mengembalikan nilai a dan b sekaligus mendapatkan hasil dari perhitungan tambah a+b.

Lalu program kita lanjutkan dengan membuat main method yang diawali dengan menuliskan baris

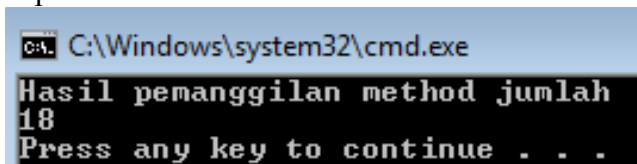
kode **public static void main(String args[])**. Di dalam main method tersebut kita menampilkan suatu kalimat atau pesan yaitu **System.out.println("Hasil pemanggilan method jumlah")**, kemudian kita panggil method jumlah yaitu dengan **System.out.println(jumlah())** sehingga muncul hasil tambah a+b yaitu 18.

PRAKTIK 5



```
Method5.java x Method4.java Method3.java Method2.java Method1.java
public class Method5{
    public static int jumlah(){
        int a = 8, b = 10;
        return (a+b);
    }
    public static void main(String args[]){
        Method5 obyek = new Method5();
        System.out.println("Hasil pemanggilan method jumlah");
        System.out.println(obyek.jumlah());
    }
}
```

output:



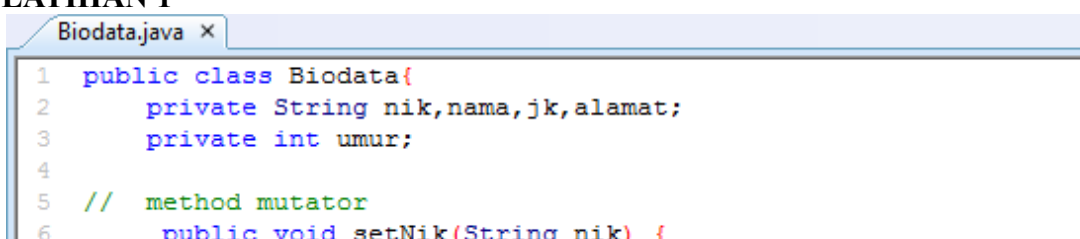
```
C:\Windows\system32\cmd.exe
Hasil pemanggilan method jumlah
18
Press any key to continue . . .
```

penjelasan:

pada program di praktik 5 ini, kita menerapkan object-oriented programming (pemrograman berorientasi objek). Untuk membuat program tersebut, kita awali dengan menuliskan nama class yaitu dengan baris kode **public class Method5**. Dibawah class langsung kita buat method bernama jumlah dengan menuliskan baris kode **public static int jumlah()**. Di dalam method tersebut kita deklarasi dua variabel bertipe data integer yaitu variabel a dengan nilai 8 dan variabel b dengan nilai 10, kedua variabel itu kita tulis dengan baris kode **int a = 8, b = 10**. Setelah itu, kedua variabel tersebut kita kembalikan nilainya dengan return. Kita tulis **return (a+b)** untuk mengembalikan nilai a dan b sekaligus mendapatkan hasil dari perhitungan tambah a+b.

Selanjutnya kita buat main method yang eksekusinya diawali dengan deklarasi **public static void main(String args[])**. Di dalamnya kita melakukan instansiasi objek (pembuatan objek) dengan menggunakan keyword new. Objek kita buat dengan syntax **Class NamaObjek = new Class()**. Jadi, pada program yang kita buat, kita instansiasi dengan menuliskan **Method5 obyek = new Method5()**. Lalu program memberikan pesan atau tulisan berupa **System.out.println("Hasil pemanggilan method jumlah")**. Terakhir kita panggil objek beserta method yaitu dengan menuliskan baris kode **System.out.println(obyek.jumlah())**.

LATIHAN 1



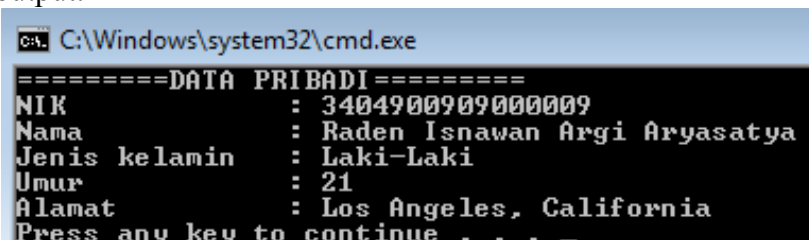
```
Biodata.java x
1 public class Biodata{
2     private String nik,nama,jk,alamat;
3     private int umur;
4
5     // method mutator
6     public void setNik(String nik) {
```

```

13         this.jk=jk;
14     }
15     public void setUmur(int umur) {
16         this.umur=umur;
17     }
18     public void setAlamat(String alamat) {
19         this.alamat=alamat;
20     }
21
22     //method accessor
23     public String getNik() {
24         return nik;
25     }
26     public String getNama() {
27         return nama;
28     }
29     public String getJk() {
30         return jk;
31     }
32     public int getUmur() {
33         return umur;
34     }
35     public String getAlamat() {
36         return alamat;
37     }
38
39     public static void main(String args[]){
40
41         Biodata data=new Biodata();
42         data.setNik("3404900909000009");
43         data.setNama("Raden Isnawan Argi Aryasatya");
44         data.setJk("Laki-Laki");
45         data.setUmur(21);
46         data.setAlamat("Los Angeles, California");
47
48         System.out.println("=====DATA PRIBADI=====");
49         System.out.println("NIK\t\t: " + data.getNik());
50         System.out.println("Nama\t\t: " + data.getNama());
51         System.out.println("Jenis kelamin\t: " + data.getJk());
52         System.out.println("Umur\t\t: " + data.getUmur());
53         System.out.println("Alamat\t\t: " + data.getAlamat());
54     }
55 }

```

output:



```

C:\Windows\system32\cmd.exe
=====DATA PRIBADI=====
NIK           : 3404900909000009
Nama          : Raden Isnawan Argi Aryasatya
Jenis kelamin : Laki-Laki
Umur          : 21
Alamat        : Los Angeles, California
Press any key to continue . . . _

```

penjelasan:

Di program ini, kita menerapkan method mutator dan accessor dan pemrograman berorientasi objek. Method mutator adalah sebuah metode yang digunakan untuk menetapkan atau mengubah nilai dari sebuah private field, skema ini biasanya ditandai dengan awalan nama “set”. Sementara method accessor adalah sebuah metode yang digunakan untuk mengembalikan nilai dari sebuah private field, skema ini biasanya ditandai dengan awalan nama “get”. Kita menggunakan 2 method

tersebut untuk menyembunyikan data dari objek. Buffer yang disediakan oleh method ini memungkinkan kita untuk mengubah data dari balik layar dan menerapkan atau mengaplikasikan validasi nilai-nilai yang telah kita atur di dalam field.

Untuk membuat program di atas, kita awali dengan mendeklarasikan class dengan menuliskan baris kode **public class Biodata**. Lalu kita buat variabel-variabel bermodifier private yang artinya tipe modifier ini mengijinkan pengaksesan class hanya dapat diakses oleh class dimana tipe ini dibuat. Variabel-variabel tersebut adalah **private String nik,nama,jk,alamat** dan **private int umur**.

Kemudian di baris 5-20 kita buat method mutator untuk menetapkan nilai suatu variabel, dan kita juga menggunakan keyword "this" untuk mengakses variabel target. Contoh penerapannya dalam program tersebut adalah:

```
public void setNik(String nik) {  
    this.nik=nik;  
}
```

di baris 22-37 kita buat method accessor untuk mengembalikan nilai field/variabel dengan menggunakan keyword "return". Method ini mengembalikan value/nilai dengan tipe data yang sama sesuai korespondensi private fieldnya, misal (String);. Contoh penerapannya dalam program tersebut adalah:

```
public String getNik() {  
    return nik;  
}
```

Kemudian kita buat main method yang eksekusinya diawali dengan **public static void main(String args[])**. Public, dapat diartikan bahwa argumen atau metode main() merupakan sebuah metode yang bersifat publik. void merupakan suatu tipe data yang menyatakan bahwa deklarasi kode tidak memerlukan nilai balik atau return. String menyatakan bahwa argumen yang terdapat pada kode java tersebut merupakan String. Args merupakan argumen bertipe data string yang mengandung array, Args dapat di misalkan sebagai memory untuk menampung program.

Di dalamnya kita lakukan instansiasi objek atau pembuatan objek bernama "data" menggunakan keyword new yang ditulis dengan baris kode **Biodata data=new Biodata()**.

kita panggil method mutator menggunakan objek data. Metode ini tidak memiliki nilai balik (return), sebab metode ini merupakan pengaturan parameter yang akan digunakan oleh koresponden private fieldnya. Penerapannya pada program di atas:

```
data.setNik("3404900909000009");  
data.setNama("Raden Isnawan Argi Aryasatya");  
data.setJk("Laki-Laki");  
data.setUmur(21);  
data.setAlamat("Los Angeles, California");
```

Terakhir kita panggil method mutator. Method ini mengembalikan value/nilai dengan tipe data yang sama sesuai korespondensi private fieldnya. Penerapannya di program tersebut:

```
System.out.println("NIK\t\t: "+ data.getNik());  
System.out.println("Nama\t\t: "+ data.getNama());  
System.out.println("Jenis kelamin\t: "+ data.getJk());  
System.out.println("Umur\t\t: "+ data.getUmur());  
System.out.println("Alamat\t\t: "+ data.getAlamat());
```

Sebelum panggil mutator, tulis **System.out.println("=====DATA PRIBADI=====")**.

LATIHAN 2

```
MinMax.java x
1  import java.util.Scanner;
2  public class MinMax{
3      public static void main(String args[]){
4          Scanner masuk = new Scanner(System.in);
5
6          int num1;
7          int num2;
8
9          System.out.print("Masukkan nilai 1 = ");
10         num1=masuk.nextInt();
11         System.out.print("Masukkan nilai 2 = ");
12         num2=masuk.nextInt();
13
14         int min = (Math.min(num1,num2));
15         int max = (Math.max(num1,num2));
16
17         System.out.println("=====");
18         System.out.println("Nilai minimal = " + min);
19         System.out.println("Nilai maksimal = " + max);
20     }
21 }
```

output:

```
C:\Windows\system32\cmd.exe
Masukkan nilai 1 = 100
Masukkan nilai 2 = 50
=====
Nilai minimal = 50
Nilai maksimal = 100
Press any key to continue . . . _
```

penjelasan:

di program ini kita menggunakan built-in method/function (method bawaan) yaitu max dan min. Method min() membandingkan nilai terkecil dari dua argumen. Method max() membandingkan nilai terbesar dari dua argumen. Untuk membuat program di atas, kita import scanner dengan baris kode **import java.util.Scanner;**. Kemudian kita deklarasikan class dengan **public class MinMax**. Dilanjutkan dengan **public static void main(String args[])**. Lalu kita masukkan scanner supaya program bisa membaca masukan dari user dan kita bisa membuat baris kode untuk input supaya user bisa memasukkan data dengan bebas dengan **Scanner masuk = new Scanner(System.in)**.

Lalu kita buat dua variabel yaitu int **num1** dan int **num2**. Supaya user bisa bebas memasukkan data angka, kita tulis baris-baris kode berikut:

```
System.out.print("Masukkan nilai 1 = ");
num1=masuk.nextInt();
System.out.print("Masukkan nilai 2 = ");
num2=masuk.nextInt();
```

kemudian kita buat dua variabel bertipe integer dengan nama min dan max. Min berguna untuk menghitung nilai terkecil dari dua nilai yang user masukkan. Max berguna untuk menghitung nilai terbesar dari dua nilai yang user masukkan. Untuk melakukan itu, kita tuliskan baris kode berikut:

```
int min = (Math.min(num1,num2));
int max = (Math.max(num1,num2));
```

terakhir, kita tampilkan nilai minimal dan maksimal dengan **System.out.println("Nilai minimal = " + min)** dan **System.out.println("Nilai maksimal = " + max)**.

TUGAS

MenuBentuk.java x

```
1  import java.util.Scanner;
2  public class MenuBentuk {
3  public static Scanner input = new Scanner(System.in);
4
5  public static void persegi()
6  {
7      double sisi,luas;
8      System.out.println("===Menghitung Luas Persegi===");
9      System.out.print("Masukkan sisi=");
10     sisi = input.nextDouble();
11     luas = sisi * sisi;
12     System.out.println("luas persegi="+luas);
13 }
14
15 public static void persegipanjang()
16 {
17     double panjang,lebar,luas;
18     System.out.println("===Menghitung Luas Persegi Panjang===");
19     System.out.print("Masukkan panjang=");
20     panjang = input.nextDouble();
21     System.out.print("Masukkan lebar=");
22     lebar = input.nextDouble();
23     luas = panjang * lebar;
24     System.out.println("luas persegi panjang="+luas);
25 }
26
27 public static void segitiga()
28 {
29     double alas,tinggi,luas;
30     System.out.println("===Menghitung Luas Segitiga===");
31     System.out.print("Masukkan alas=");
32     alas = input.nextDouble();
33     System.out.print("Masukkan tinggi=");
34     tinggi = input.nextDouble();
35     luas = (alas * tinggi)/2;
36     System.out.println("luas persegi segitiga="+luas);
37 }
38 }
39
40 public static void main(String[] args) {
41     int pilih;
42
43     do
44     {
45         System.out.println("=====MENU=====");
46         System.out.println("1.Luas Persegi");
47         System.out.println("2.Luas Persegi Panjang");
48         System.out.println("3.Luas Segitiga");
49         System.out.println("0.Keluar");
50         System.out.print("Masukkan Pilihan:");
51
52         pilih =input.nextInt();
53         switch(pilih)
54         {
55             case 1 : persegi();
56                 break;
57             case 2 : persegipanjang();
58                 break;
59             case 3 : segitiga();
60                 break;
61             }
62         }
63     while ((pilih!=0) || (pilih>3));
64     System.out.println("Terima Kasih");
65 }
66 }
```

output:

```
C:\Windows\system32\cmd.exe
=====MENU=====
1.Luas Persegi
2.Luas Persegi Panjang
3.Luas Segitiga
0.Keluar
Masukkan Pilihan:1
===Menghitung Luas Persegi===
Masukkan sisi=8
luas persegi=64.0
=====MENU=====
1.Luas Persegi
2.Luas Persegi Panjang
3.Luas Segitiga
0.Keluar
Masukkan Pilihan:2
===Menghitung Luas Persegi Panjang===
Masukkan panjang=11
Masukkan lebar=8
luas persegi panjang=88.0
=====MENU=====
1.Luas Persegi
2.Luas Persegi Panjang
3.Luas Segitiga
0.Keluar
Masukkan Pilihan:3
===Menghitung Luas Segitiga===
Masukkan alas=6
Masukkan tinggi=12
luas persegi segitiga=36.0
=====MENU=====
1.Luas Persegi
2.Luas Persegi Panjang
3.Luas Segitiga
0.Keluar
Masukkan Pilihan:0
Terima Kasih
Press any key to continue . . . _
```

penjelasan:

pertama, kita deklarasikan class yaitu dengan baris kode **public class MenuBentuk**.

Method pertama yang kita buat adalah **public static void persegi()**. Di dalamnya user memasukkan data sisi di field **System.out.print("Masukkan sisi=")** dengan kode **sisi = input.nextDouble();**. Lalu kita buat variabel bernama luas sebagai perhitungan luas persegi yaitu **luas = sisi * sisi;**. Kemudian, kita tampilkan hasil perhitungan dengan **System.out.println("luas persegi="+luas)**.

Method kedua yang kita buat adalah **public static void persegipanjang()**. Di dalamnya user memasukkan data panjang di field **System.out.print("Masukkan panjang=")** dengan kode **panjang = input.nextDouble();** dan data lebar di field **System.out.print("Masukkan lebar=")** dengan kode **lebar = input.nextDouble()**. Lalu kita buat variabel bernama luas sebagai perhitungan luas persegi panjang yaitu **luas = panjang * lebar**. Kemudian, kita tampilkan hasil perhitungan dengan **System.out.println("luas persegi panjang="+luas)**.

Method ketiga yang kita buat adalah **public static void segitiga()**. Di dalamnya user memasukkan data alas di field **System.out.print("Masukkan alas=")** dengan kode **alas = input.nextDouble();** dan data tinggi di field **tinggi = input.nextDouble()** dengan kode **tinggi = input.nextDouble()**. Lalu kita buat variabel bernama luas sebagai perhitungan luas segitiga yaitu **luas = (alas * tinggi)/2**. Kemudian, kita tampilkan hasil perhitungan dengan **System.out.println("luas persegi panjang="+luas)**.

Kemudian kita buat main method yang diawali dengan **public static void main(String[] args) {**.

kita deklarasikan variable integer bernama pilih. Kemudian kita lakukan perulangan do while. Perulangan ini melakukan eksekusi menampilkan output terlebih dahulu sebelum melakukan proses perulangan. Pertama kita tulis do dan kita buat menu program dengan menuliskan:

```
System.out.println("=====MENU=====");
System.out.println("1.Luas Persegi");
System.out.println("2.Luas Persegi Panjang");
System.out.println("3.Luas Segitiga");
System.out.println("0.Keluar");
System.out.print("Masukkan Pilihan:");
```

kemudian kita lakukan switch case. Kita buat switch(pilih) dan jika case 1 maka yang terpilih adalah method persegi(), jika case 2 maka yang terpilih adalah method persegipanjang(), jika case 3 maka yang terpilih adalah method segitiga(). Kode untuk menuliskan switch case tersebut adalah:

```
switch(pilih)
{
    case 1 : persegi();
            break;
    case 2 : persegipanjang();
            break;
    case 3 : segitiga();
            break;
```

jika kita memilih selain 1,2,dan 3 maka muncul pesan "Terima Kasih". Kodenya adalah:

```
while ((pilih!=0)|| (pilih>3));
System.out.println("Terima Kasih");
```

=====

KESIMPULAN

Di pertemuan ke-8 ini saya berhasil dapat memahami, membuat dan menyelesaikan kasus dengan menggunakan method tanpa parameter. Ada bermacam-macam jenis method yang saya pelajari di pertemuan ke-8 ini, yaitu method void, method non void, method accessor, method mutator, hingga method bawaan yaitu min dan max. Terakhir, di bagian Tugas saya membuat program method berupa menu bangun persegi, persegi panjang, segitiga, serta menggunakan perulangan do while sebagai cara untuk memilih pilihan di dalam menu yang sudah dibuat.