

Method

Teguh Sutanto

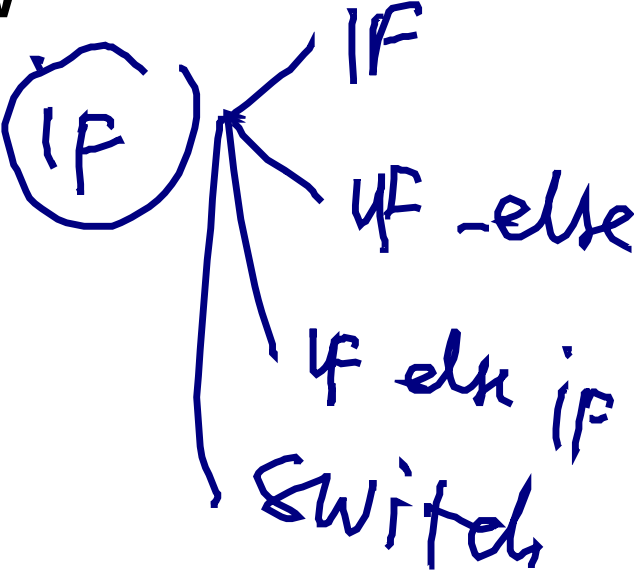
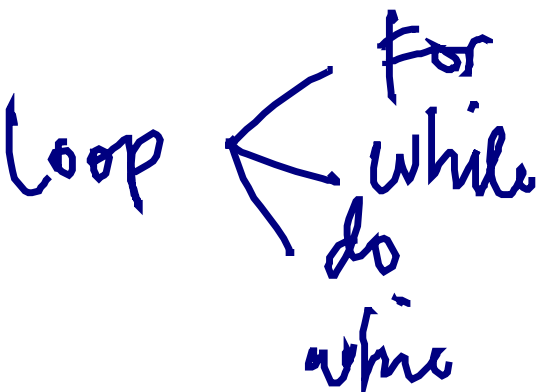
teguh.sutanto@gmail.com

Tujuan

- Memahami method ✓
- Membuat Method ✓
 - 1 Return Value
 - 2 Void
 - A Static
 - B Non Static
 - Parameter

```
public class TestNoMethod{  
    [redacted] ✓  
    [redacted] ✓  
    [redacted] ✓  
    [redacted] ✓  
    public static void main(String []argv){  
    }  
}
```

Review

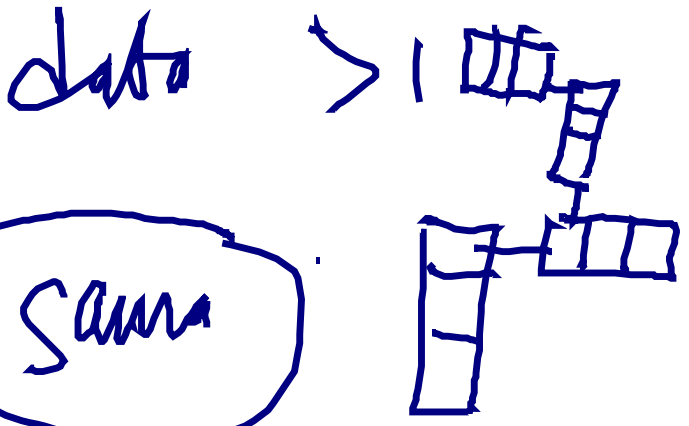


Array

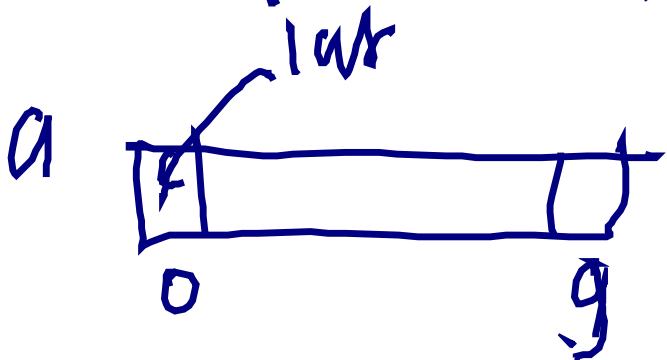
1 ←
length+1

def

Var → Menyimpan

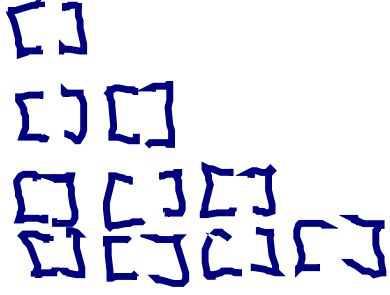


```
int [] a = new int [10];
```



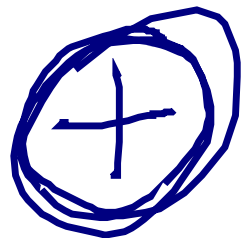
Sama

1 Dim
n dimensi



String

equals
equalsIgnoreCase



Menambahkan 2 buah string ke

```
String str = "Java";  
String str2 = "Programming";  
String str3 = str + str2;
```

length()

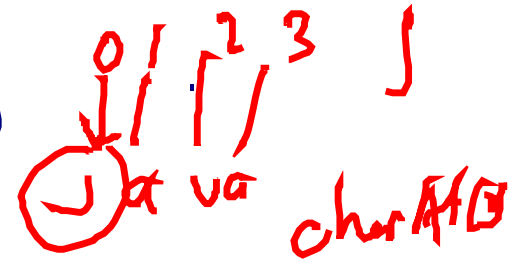
charAt()

substring()
substring(n, m)

str2.charAt(0);

```
if (str == "Java")
```

```
if (str.equals("Java"))
```



Next

- Method
- Class
- Using Util Class
 - Vector
- Case Study

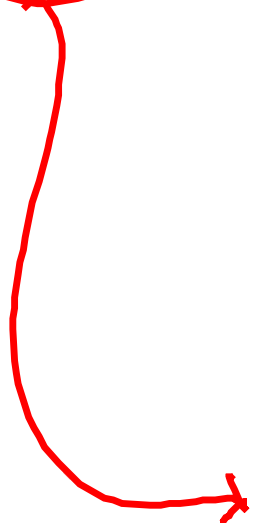
was method

Method

if

a++

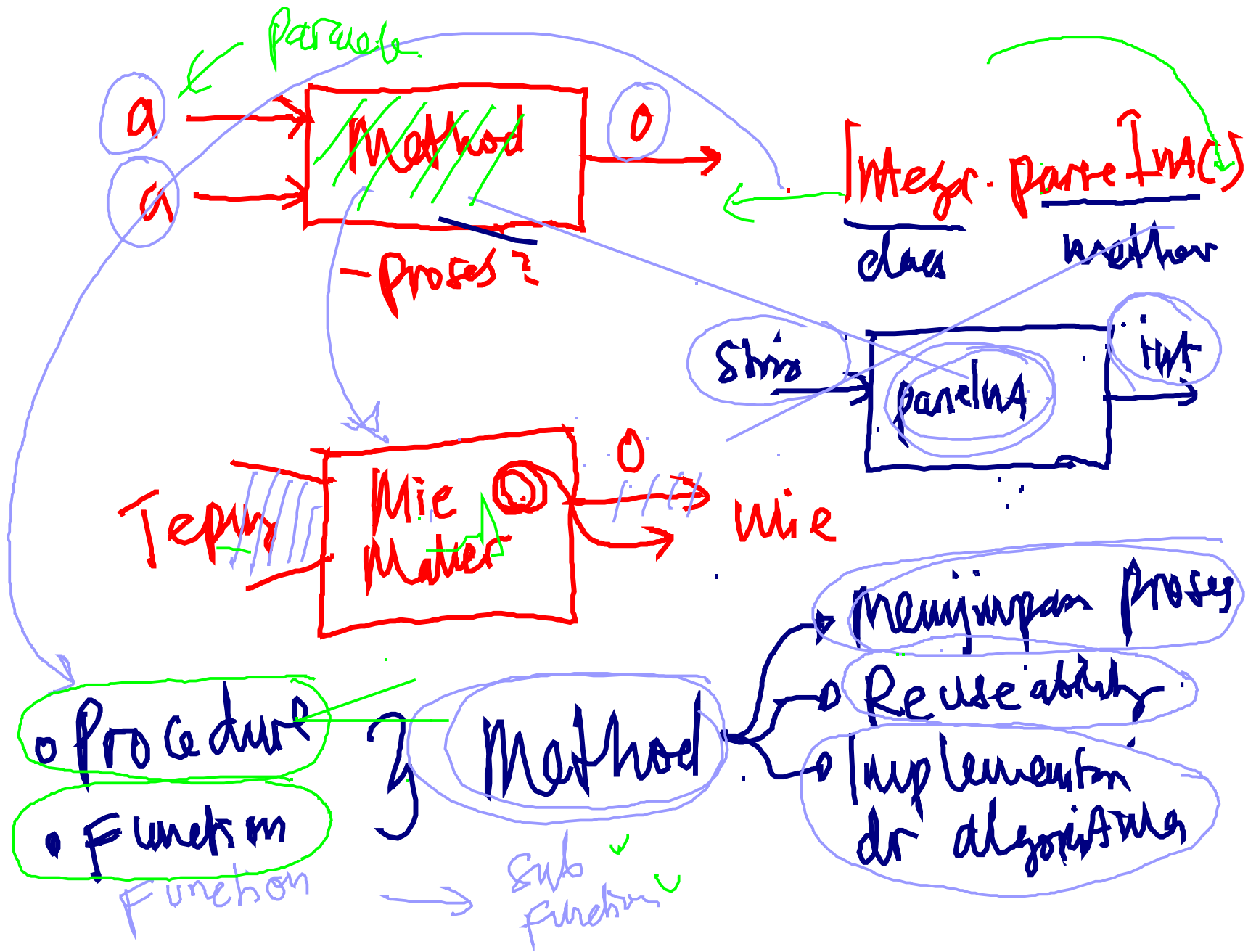
Semua proses di lakukan
dalam sebuah method
exc - Declarasi variable



```
main () {
```

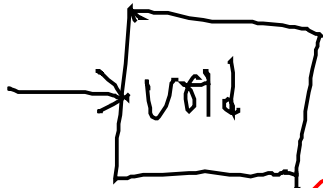
```
    int a = 10;  
    a++;
```

```
}
```



Type Method

by Return Value



1

Void

← tanpa return value
contoh (String str) {}

void
3

← return value sesuai
dengan tipe data yg
di deklarasikan.

2

Data Type



int add (int a,
int b) {}

parameter

3

```
//return value int
public static int add(int a, int b){
    return a+b;
}
```

return value
nama
method

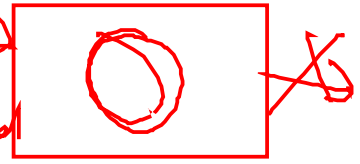
```
//tanpa return value - void
public static void cetak(String str){
    System.out.println(str);
}
```

```
public static void main(String []argv){
    int bil1=100;
    int bil2=20;
    int hasil=add(bil1,bil2);
    cetak("hasil = " + hasil);
}
```

100 20

hasil ada
return
nama
method

input



a = bil1

add(bil2, bil1)

a = bil2

b = bil1

120

hasil = 120

```
public static void cetakGaris(){
    System.out.println("*****");
}
public static void cetakGaris(int
pjGaris){//loop
    int i=0;
    while(i<pjGaris){
        System.out.print("*");
        i++;
    }
    System.out.println();
}
```

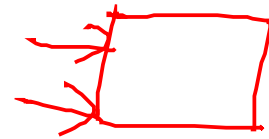
```
int add (int a, int b, int c) {  
    c = a + b;  
    return c;  
}
```

3



Anonymous Method

```
String str = "Java";
```



Technical

nameMethod ([params])

- void ✓
- datatype ✓

Unique

in 1 class

1 class
nameMethod

two class
NameClass.nameMeth.
nameObj.nameMeth.

Math.random() ✓
Integer.parseInt() ✓

Pelajaran
Method

public
private

protected
package
return.

void
type

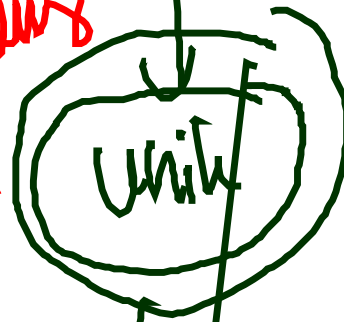
nama Method ([param]) {}

access static
modifier

return value

3

Overloading
Method



Cetak Garis (0);

Cetak Garis (10);

Cetak Garis (5, "#");

nama sama tetapi
jumlah parameter
beda

Latihan 1 (LatMethod01.java)

- Buatlah method untuk mencetak garis dengan pola "*" sebanyak n kali

Contoh:

```
cetakGaris(5); //*****
```

```
cetakGaris(3); //***
```

```
cetakGaris(10); //*****
```

Latihan 2 (lanjutan...)

- Tambahkan method cetakGaris() dengan parameter panjang garis berupa integer dan pola garis berupa String
- contoh:
 - cetakGaris(3,"#");//###
 - cetakGaris(4,"@");//@@@@
 - cetakGaris(5,"\$");//\$\$\$\$\$

LatMethod02

```
public class LatMethod02{
    public static void cetakBanner(String kata){
        int pjGaris=kata.length()+2;
        LatMethod01.cetakGaris(pjGaris);
        System.out.println("*" + kata + "*");
        LatMethod01.cetakGaris(pjGaris);
    }
    public static void cetakBanner2(String kata){
        //
    }
}
```

Review Hari ini

- Method
 - void
 - data type
- Deklarasi Method
- Method Overloading

Latihan Di Rumah

- Buatlah class dengan nama OperasiArray yang memiliki method sebagai berikut:

Nama Method	Keterangan
<code>cetak(int []data)</code>	Mencetak data array
<code>hitungTotal(int []data)</code>	Menghitung total elemen array
<code>nilaiMax(int []data)</code>	mencari nilai terbesar
<code>nilaiMin(int []data)</code>	mencari nilai terkecil
<code>cari(int []data, int key)</code>	mencari data
<code>rataRata(int []data)</code>	menghitung rata-rata

int hitungTotal()

```
public static int hitungTotal(int []data){  
    int total=0;  
    for(int i=0;i<data.length;i++)  
        total+=data[i];  
    return total;  
}
```

-

int nilaiMax()

```
public static int nilaiMax(int []data){  
    int max=data[0];  
    for(int i=0;i<data.length;i++)  
        if(max<data[i])  
            max=data[i];  
    return max;  
}
```