

Method

Teguh Sutanto
teguh.sutanto@gmail.com

Tujuan

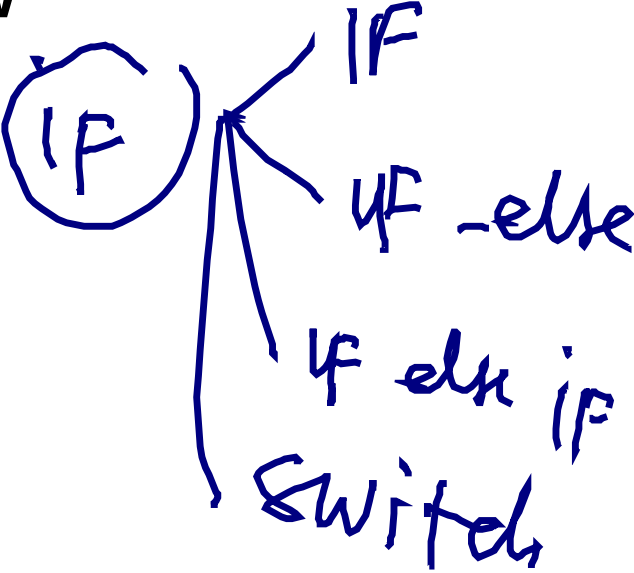
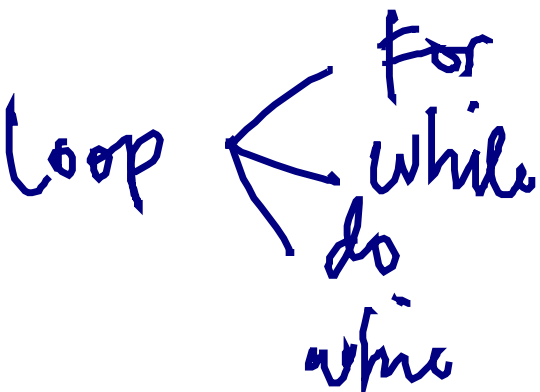
- Memahami method
- Membuat Method
 - Return Value
 - Void
 - Static
 - Non Static
 - Parameter

```
public class NoMethod {  
    int a = 100; ✓  
    a++; // 100 ✓  
    if (a >= 100) ✓  
        System.out.println(a); ✓  
    public static void main() ✓  
}
```

3

Salah benar

Review

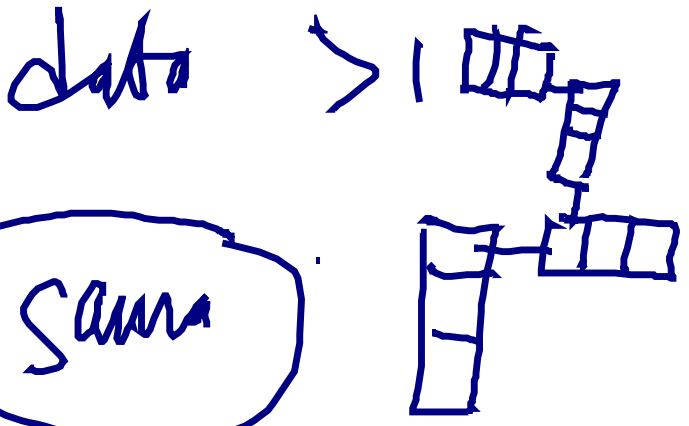


Array

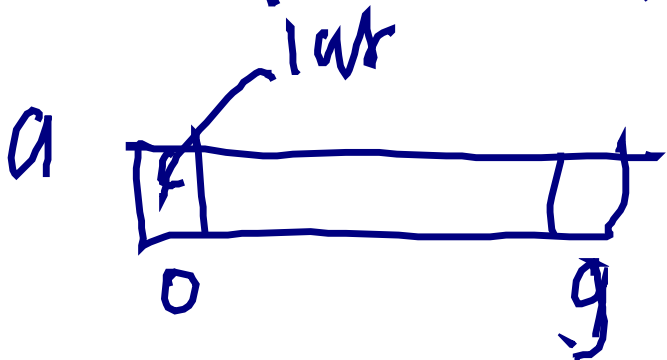
1 ←
length+1

def

Var → Menyimpan

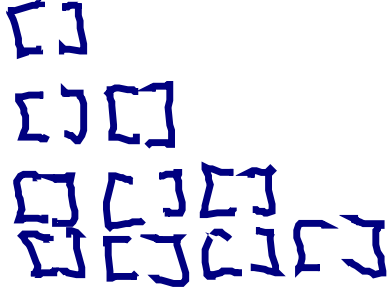


```
int [] a = new int [10];
```



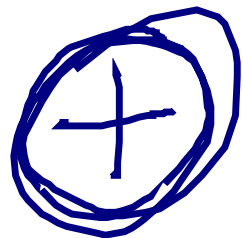
Sama

1 Dim
n dimensi



String

equals
equalsIgnoreCase



→ Menambahkan

2 buah string ke

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

```
String str2 = "Programming";
```

```
String str3 = str + str2;
```

```
str2.charAt(0);
```

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

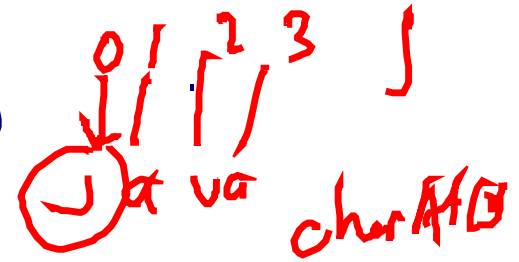
```
if (str.equals("Java"))
```

length()

charAt()

substring()

substring(n, m)



Next

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

was method

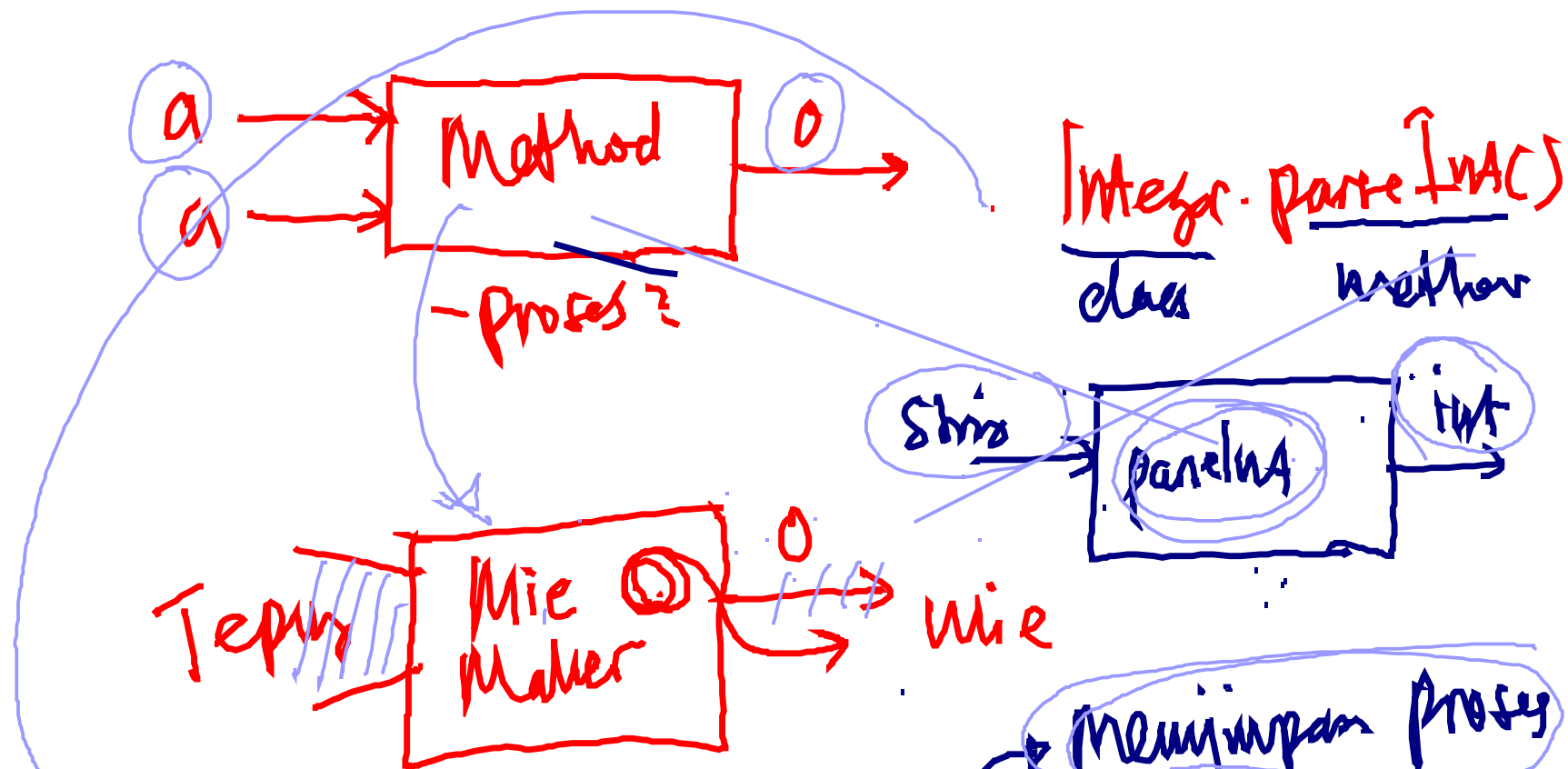
Method

if

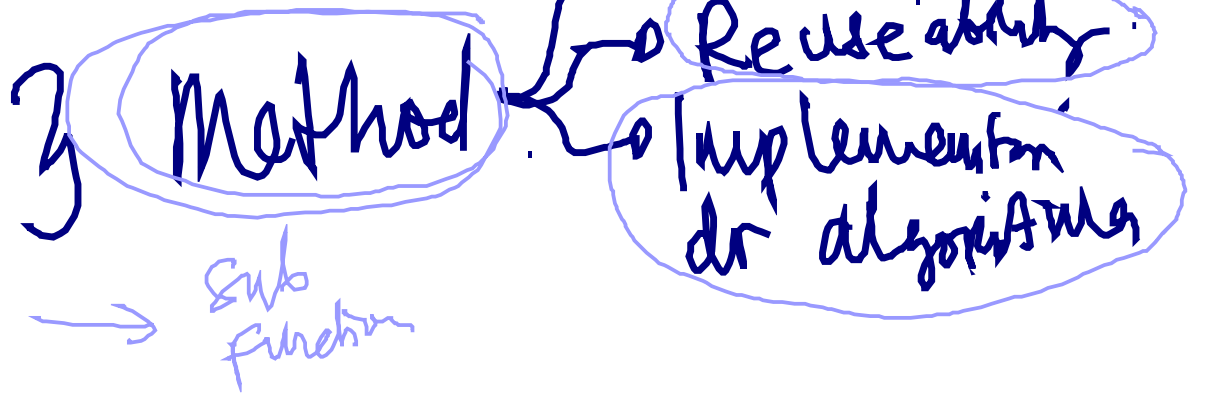
a++

Semua proses di lakukan
dalam sebuah method
exc - Declaration variable

```
main() {  
    int a = 10;  
    a++;  
}
```



- o procedure
- Function
- Function



Type Method by Return Value



1

Void

Void

3

tanpa Return Value
contoh (String, String)

2

Data Type

Return Value sesuai
dan type data yg
di deklarasi kan.

parameter



int add (int a,
int b) {

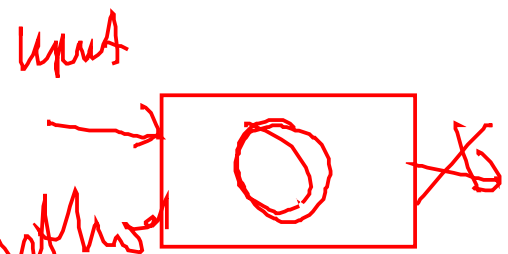
}

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

return value
nama
method

100 20
return value

nama
method



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();
}
```

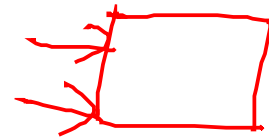
```
int add (int a, int b, int c) {  
    c = a + b;  
    return c;  
}
```

3



Anatomy Method

String str = "java";



str.equals
↑
obj
↑
method



- ↳ void
- ↳ Duplicate

Math.random()
Integer.parseInt()

↓
Unique
↓
in 1 class

↳ class
name Method

↳ two class
name class - name method
name obj - name method

Pelajaran
Method

public
private

protected
package
return.

void
type

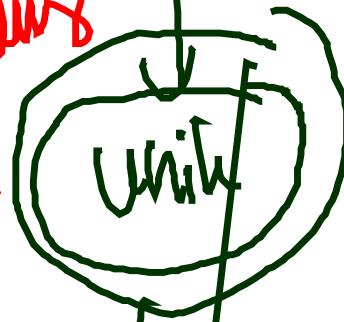
nama Method ([param]) {}

access static
modifier

return value

3

Overloading
Method



Nama Sama tetapi
Jumlah Parameter
Berbeda

Cetak Garis (0);

Cetak Garis (10);

Cetak Garis (5, "#");

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);  
    }  
}
```

Buatlah method
cetakBanner2(String str)

- `acr_i @_l l cp0&F cjjm'9`

Review Hari ini

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

Latihan Di Rumah

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

L_k_Kcf nb	l crcp e
acr_i & rYb_r'	Kcl acr_i b_r_pp_w
f gsl eRmj & rYb_r'	Kcl ef gsl emj cjck cl_pp_w
l g_gK_v & rYb_r'	kcl a_pgl g_grcp`cq_p
l g_gK_d & rYb_r'	kcl a_pgl g_s rcp_i cagj
a_pgl rYb_r *g r i cw	kcl a_pgl b_r
pr_P_r & rYb_r'	kcl ef gsl epr_+pr_

