

Algoritma Pemrograman 1

Laboratorium Informatika Universitas Gunadarma

Pertemuan 3:

Object, Method, Class, Constructor, dan Exception

Method: Definisi

- Method(metode) adalah sekumpulan statement program yang di satukan menjadi sebuah subprogram atau fungsi, diawali dengan tanda “{“ diakhiri dengan tanda “}”.
- Jenis method ada dua yaitu
 1. Method tidak membalikkan nilai
 2. Method membalikkan nilai

Method: Bentuk Umum

- Bentuk umum dalam mendeklarasikan suatu method adalah seperti berikut ini:

```
TipeMethod namaMethod(tipeData namaParameter) {  
    //statemen;  
}
```

Method: Jenis

- **Metode tidak membalikkan nilai**

Jika diberi awalan dengan kata void maka method tersebut tidak memberi nilai balik.

contoh:

```
public void Namamethod(){  
    System.out.println("INI METHOD TANPA NILAI BALIK");  
}
```

- **Method membalikkan nilai.**

Method diberi awalan sebuah tipe data maka method tersebut akan memberi nilai balik berupa data yang bertipedata sama dengan method tersebut.

contoh :

```
public int Namamethod(){  
    int nilai = 10;  
    return nilai;  
}
```

Method: Jenis Lain

- **Method berparameter**

Method berparameter adalah method yang dideklarasikan dengan memiliki satu atau beberapa nilai.

Contoh:

```
void Namamethod(String a) {  
    System.out.println("a adalah: "+a);  
}
```

Class dan Object: Definisi

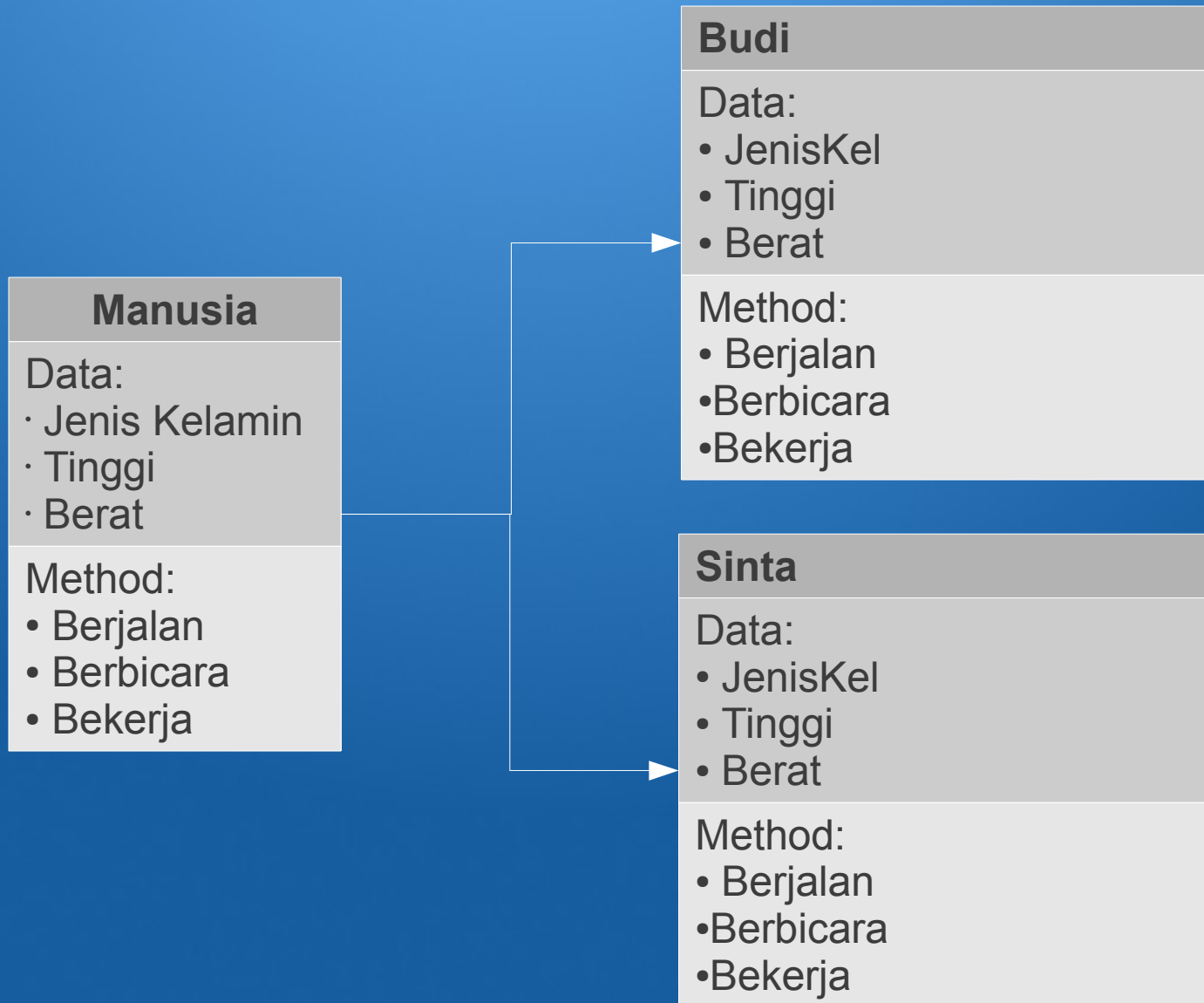
- Class(kelas) adalah blue-print atau kerangka yang mendefinisikan variabel, dan method yang akan digunakan oleh object.

Contoh: Manusia, Mobil, Hewan Herbivora

- Object(objek) adalah bentuk nyata dari suatu kelas. Objek dapat menggunakan berbagai variabel dan method yang didefinisikan pada suatu kelas.

Contoh: Objek manusia(Budi, Sinta, Banu), Objek mobil(Avanza, Rush, Kijang)

Class dan Object: Contoh



Class dan Object: Bentuk Umum

- Bentuk umum dari class dan object adalah seperti berikut ini;

```
class namaClass{
    tipeData var1;
    tipeData varN;
    tipe method1(tipe parameter){
        //statemen dalam method;
    }
    Tipe method2(tipe parameter0{
        //statemen dalam method;
    }
}
```

Class dan Object: Penggunaan

- Untuk menggunakan class yang telah didefinisikan maka terlebih dahulu kita membuat objek dari kelas tersebut dengan cara:

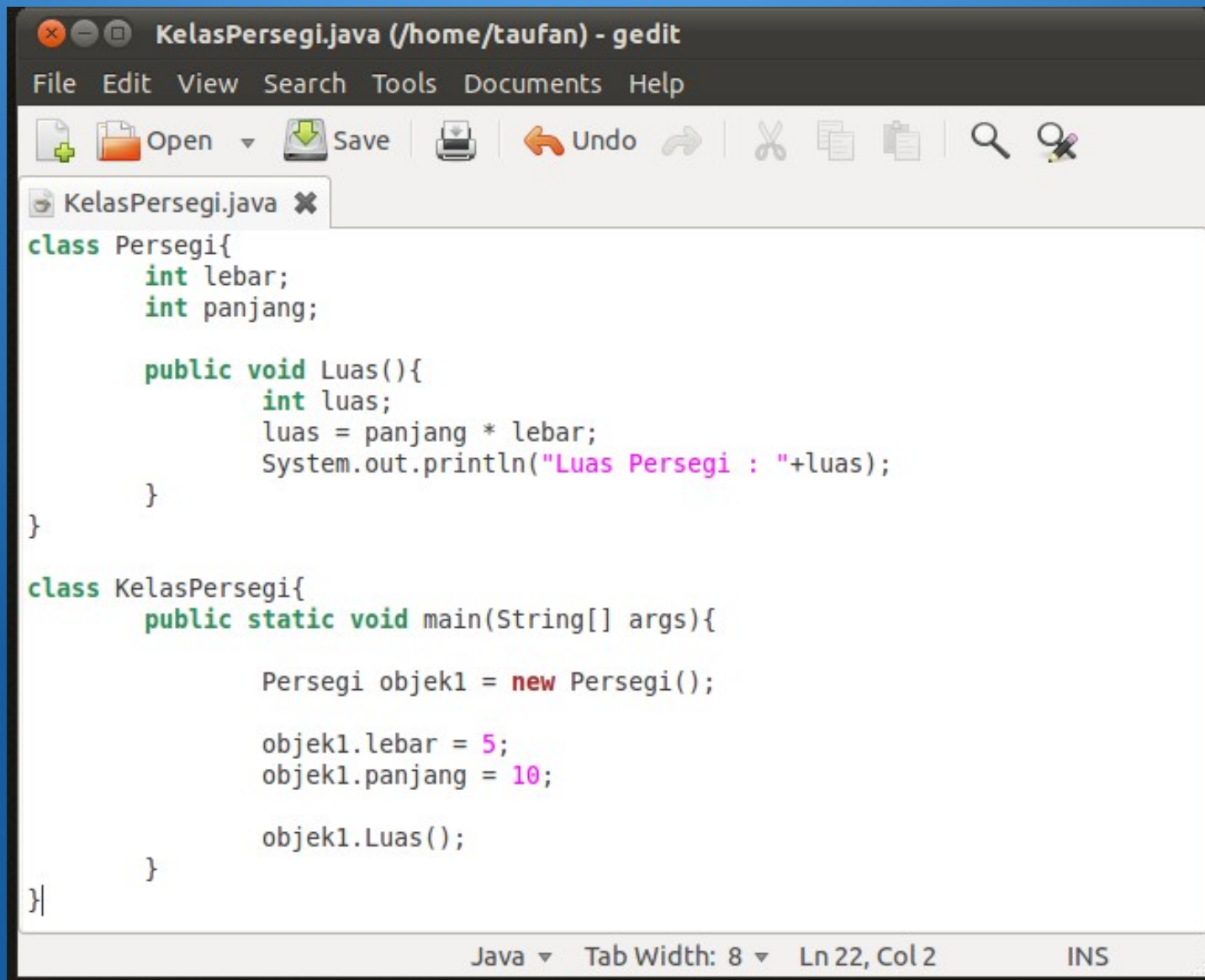
```
NamaKelas namaObjek;
```

```
NamaObjek = new NamaKelas();
```

Atau,

```
NamaKelas namaObjek = new NamaKelas();
```

Class dan Object: Contoh



```
KelasPersegi.java (/home/taufan) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
KelasPersegi.java x
class Persegi{
    int lebar;
    int panjang;

    public void Luas(){
        int luas;
        luas = panjang * lebar;
        System.out.println("Luas Persegi : "+luas);
    }
}

class KelasPersegi{
    public static void main(String[] args){

        Persegi objek1 = new Persegi();

        objek1.lebar = 5;
        objek1.panjang = 10;

        objek1.Luas();
    }
}

Java Tab Width: 8 Ln 22, Col 2 INS
```

Constructor: Definisi

- Konstruktor adalah suatu metode yang dapat digunakan untuk memberi nilai awal pada saat objek diciptakan. Konstruktor akan dipanggil secara otomatis begitu objek diciptakan.
- Konstruktor memiliki ciri :
 1. Namanya sama dengan nama kelas
 2. Tidak memiliki nilai balik(dan juga tidak boleh ada kata void didepannya)

Constructor: Bentuk Umum

- Bentuk umum untuk mendeklarasikan constructor adalah seperti berikut ini:

```
class namaClass{  
    NamaClass() {  
        //statemen;  
    }  
}
```

Constructor: Contoh

```
KelasPersegi.java (/home/taufan) - gedit
File Edit View Search Tools Documents Help
Open Save Undo
KelasPersegi.java
class Persegi{
    int lebar;
    int panjang;

    Persegi(int l,int p){
        lebar = l;
        panjang = p;
    }

    public void Luas(){
        int luas;
        luas = panjang * lebar;
        System.out.println("Luas Persegi : "+luas);
    }
}

class KelasPersegi{
    public static void main(String[] args){

        Persegi objek1 = new Persegi(5,10);
        objek1.Luas();
    }
}

Java Tab Width: 8 Ln 20, Col 52 INS
```

Exception: Definisi

- Exception dimunculkan pada program saat menemukan keadaan yang dapat menyebabkan program menjadi error. Sebuah Exception akan menyebabkan aliran program keluar dari aliran yang normal. Penanganan eksepsi atau kesalahan pada java, ada dua cara :
 1. Menangkap eksepsi
 2. Melemparkan Eksepsi

Exception: Jenis

- **Menangkap eksepsi**

Kesalahan atau eksepsi dapat dikendalikan dengan cara “ditangkap”. Untuk menangkap suatu eksepsi java menyediakan statement try dan catch. Bentuk umum dari penangkapan eksepsi adalah sebagai berikut:

- **Bentuk umum:**

```
Try{  
    // Blok yang diperkirakan terdapat kesalahan  
}catch(JenisException var){  
    //blok yang akan di jalankan bila terjadi error  
}
```


Exception: Jenis

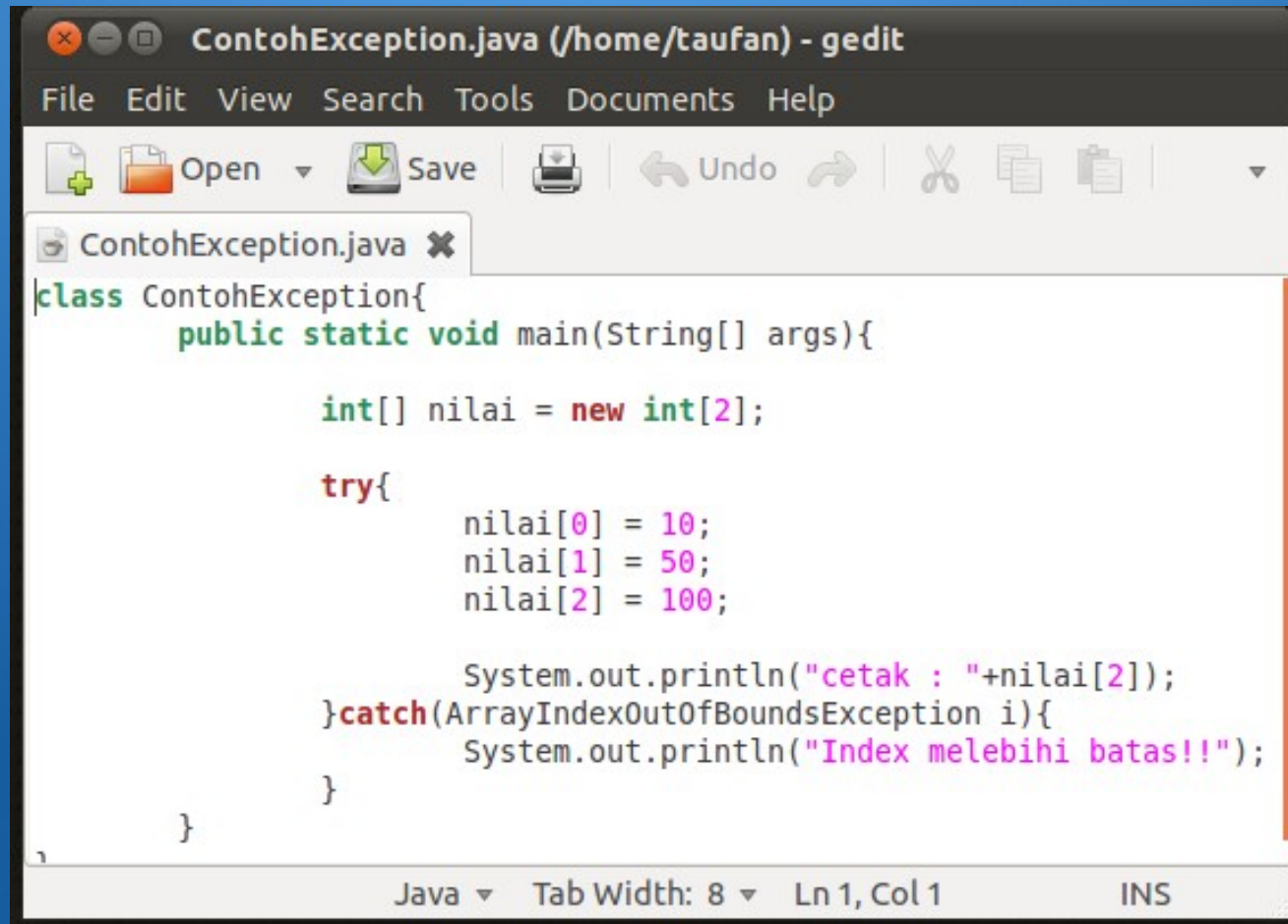
- **Melemparkan Eksepsi**

Selain menggunakan try untuk mengendalikan eksepsi java juga menyediakan pernyataan yang digunakan untuk melemparkan eksepsi yaitu pernyataan throws. Bentuk umum dari pernyataan throws dalam metode adalah:

- **Bentuk Umum:**

```
public namaMethod/Class () throws jenisExcepsi{  
    //Blok yang diperkirakan terdapat kesalahan  
}
```

Exception: Contoh



```
ContohException.java (/home/taufan) - gedit
File Edit View Search Tools Documents Help
+ Open Save Print Undo
ContohException.java
class ContohException{
    public static void main(String[] args){

        int[] nilai = new int[2];

        try{
            nilai[0] = 10;
            nilai[1] = 50;
            nilai[2] = 100;

            System.out.println("cetak : "+nilai[2]);
        }catch(ArrayIndexOutOfBoundsException i){
            System.out.println("Index melebihi batas!!");
        }
    }
}
Java Tab Width: 8 Ln 1, Col 1 INS
```

Laporan Akhir

- Modifikasi file KelasPersegi.java sehingga dapat melakukan perhitungan luas segitiga?

Laporan Pendahuluan Pertemuan 4

1. Tuliskan bentuk umum sintaks atau method untuk penanganan String berikut:
 - Membuat objek String
 - Menggabungkan String
 - Membandingkan dan mengurutkan String
 - Mendapatkan panjang String
 - Mendapatkan karakter dari String
2. Bagaimana cara membuat objek file!
3. Sebutkan macam-macam operasi file serta apa perbedaan dari operasi file tersebut.

THANKS.