



Riza Satria Perdana, S.T., M.T.

Teknik Informatika - STEI ITB

Initializing Field

Initializing Field

Pemrograman Berorientasi Objek



Static Initialization Block

- Blok kode yang diapit { } dan didahului keyword **static**
- Kelas boleh memiliki blok ini lebih dari satu

```
static {  
    // whatever code is needed for initialization goes here  
}
```



Private Static Methode

- Alternatif dari static block
- Kelebihannya dapat di-reuse bila diperlukan (di-call lagi)

```
class Whatever {  
    public static varType myVar = initializeClassVariable();  
  
    private static varType initializeClassVariable() {  
  
        //initialization code goes here  
  
    }  
}
```

Inisialisasi Instance Member

- Biasanya kode inisialisasi dituliskan di konstruktor
- Ada alternatif lain:
 - Initializer Block
 - Menggunakan Final Methode

Initializer Block

- Initializer Block untuk instance variabel mirip dengan static initializer block tapi tanpa keyword **static**
- Java compiler akan meng-copy initializer block ke setiap konstruktor

```
{  
    // whatever code is needed for initialization goes here  
}
```

Final Methode

- Final Methode tidak bisa di-override di sub class
- Berikut contoh final metode untuk menginisialisasi instance variabel

```
class Whatever {  
    private varType myVar = initializeInstanceVariable();  
  
    protected final varType initializeInstanceVariable() {  
  
        //initialization code goes here  
  
    }  
}
```

Terima Kasih