

Tipe Data

Java merupakan bahasa pemrograman yang *strongly typed*, maka kita tidak bisa mengabaikan tipe data. Kita harus tahu data seperti apa yang disimpan ke dalam variabel. Selain itu, Java juga bersifat *statically typed*, yang artinya setiap variabel harus dideklarasikan terlebih dahulu sebelum digunakan.

Setiap bahasa pemrograman memiliki aturan tersendiri dalam menuliskan sebuah variabel. Begitupun pada Java. Setiap variabel harus dideklarasikan terlebih dahulu tipe datanya. Tipe data digunakan untuk menentukan jenis nilai yang akan ditampung pada variabel tersebut. Terdapat 2 (dua) jenis tipe data pada Java, yaitu tipe data *primitive/primitif* dan *reference/referensi*. Mari kita pelajari masing-masing komponen-komponen di dalam keduanya.

Tipe Data Primitif

Tipe data primitif adalah tipe data standar yang tidak diturunkan dari objek manapun. Tipe data primitif telah ditentukan dengan kata kuncinya masing-masing. Terdapat 8 (delapan) tipe data primitif yang Java dukung, antara lain:

1. Byte

Tipe data integer 8 bit yang digunakan untuk menampung angka dalam range yang kecil. Nilai minimum dan maksimum dari tipe data ini adalah -128, dan 127. Tipe data byte memiliki nilai *default* 0. Ia digunakan untuk menghemat ruang dalam *array* yang besar, terutama pada bilangan bulat, karena 1 (satu) byte 4 (empat) kali lebih kecil dari tipe data int.

```
1. byte value = 10;
2. byte anotherValue = -10;
```

2. Short

Merupakan tipe data integer 16 bit yang digunakan untuk menampung angka dalam *range* menengah, yaitu antara -32,768 sampai 32,767. Sama seperti tipe sebelumnya, nilai *default*-nya 0. Ia bisa digunakan untuk menghemat memori seperti tipe data byte namun hanya 2 (dua) kali lebih kecil dari tipe data int.

```
1. short value = 15000;
2. short anotherValue = -20000;
```

3. Int

Merupakan tipe data integer 32 bit yang digunakan untuk menyimpan angka dalam *range* cukup besar, yakni antara -2,147,483,648 sampai 2,147,483,647 dengan nilai *default* 0. Jika kita tidak memperhatikan penggunaan memori, tipe data inilah yang biasanya dipakai.

- ```
1. int value = 150000;
2. int anotherValue = -200000;
```

#### 4. Long

Merupakan tipe data integer yang lebih besar jika dibandingkan dengan tipe data int. Ukurannya 64 bit dan bisa digunakan untuk menyimpan angka dengan range antara -9,223,372,036,854,775,808 sampai 9,223,372,036,854,775,807. Tipe data long memiliki nilai *default* 0L.

- ```
1. long value = 150000L;
2. long anotherValue = -200000L;
```

5. Float

Merupakan sebuah tipe data yang bisa digunakan untuk menampung angka desimal. Nilai *default*-nya 0.0f.

- ```
1. float value = 3.5f;
```

#### 6. Double

Merupakan sebuah tipe data yang mirip seperti tipe data float, namun memiliki kapasitas yang lebih besar. Nilai *default*-nya 0.0d.

- ```
1. double value = 5.0;
```

7. Boolean

Merupakan sebuah tipe data yang memiliki 2 (dua) macam nilai, yaitu true dan false. Nilai *default*-nya false.

- ```
1. boolean value = true;
2. boolean anotherValue = false;
```

#### 8. Char

Merupakan sebuah tipe data yang bisa digunakan untuk menampung karakter. Nilai karakter tersebut dibungkus di dalam tanda ' '.

- ```
1. char item = 'A'
```

Tipe Data Reference

Tipe data *reference* merupakan sebuah tipe data yang merujuk ke sebuah objek atau *instance* dari sebuah *class*. Salah satu tipe data yang termasuk ke dalam tipe data *reference* adalah string. Tipe data string menunjuk ke *instance* dari *class* `java.lang.String`.

Selain itu, kita bisa membuat tipe data sendiri. Sebagai contoh, ketika kita memiliki sebuah kelas `User`, kita bisa membuat variabel baru dengan tipe data *class* `User` tersebut. Kita perlu membuat *instance* dari kelas tersebut:



```
1. User user = new User();
```

Dengan begitu kita bisa mengakses semua atribut dan fungsi yang ada pada kelas **User** menggunakan variabel **user**.

[< Sebelumnya](#)[Selanjutnya >](#)