



POLITEKNIK ELEKTRONIKA NEGERI SURABAYA



DIKTISAINTEK
BERDAMPAK

KAMPUS
INOVASI

ARRAY

Pemrograman Berorientasi Objek

Yunia Ikawati

Teknik Informatika-PENS

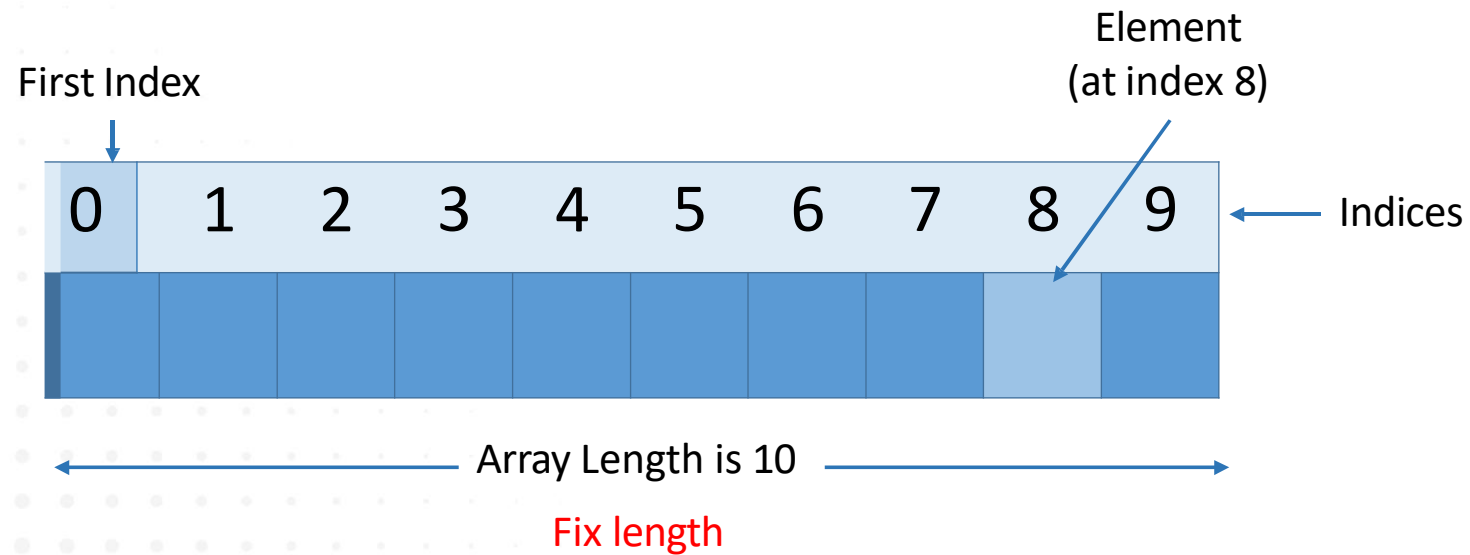


Overview Array

- Array adalah :
 - Object yang berfungsi sebagai **kontainer**
 - Menampung **sejumlah** data
 - Satu tipe yang **sama**



Overview Array



Declaring Array

- Cara mendeklarasikan array:
`tipe_array nama_array[];`
`tipe_array[] nama_array;`
- Contoh :
`int nilai[];`
`int [] nilai;`
`char [] huruf;`
`char huruf [];`

Declaring Array

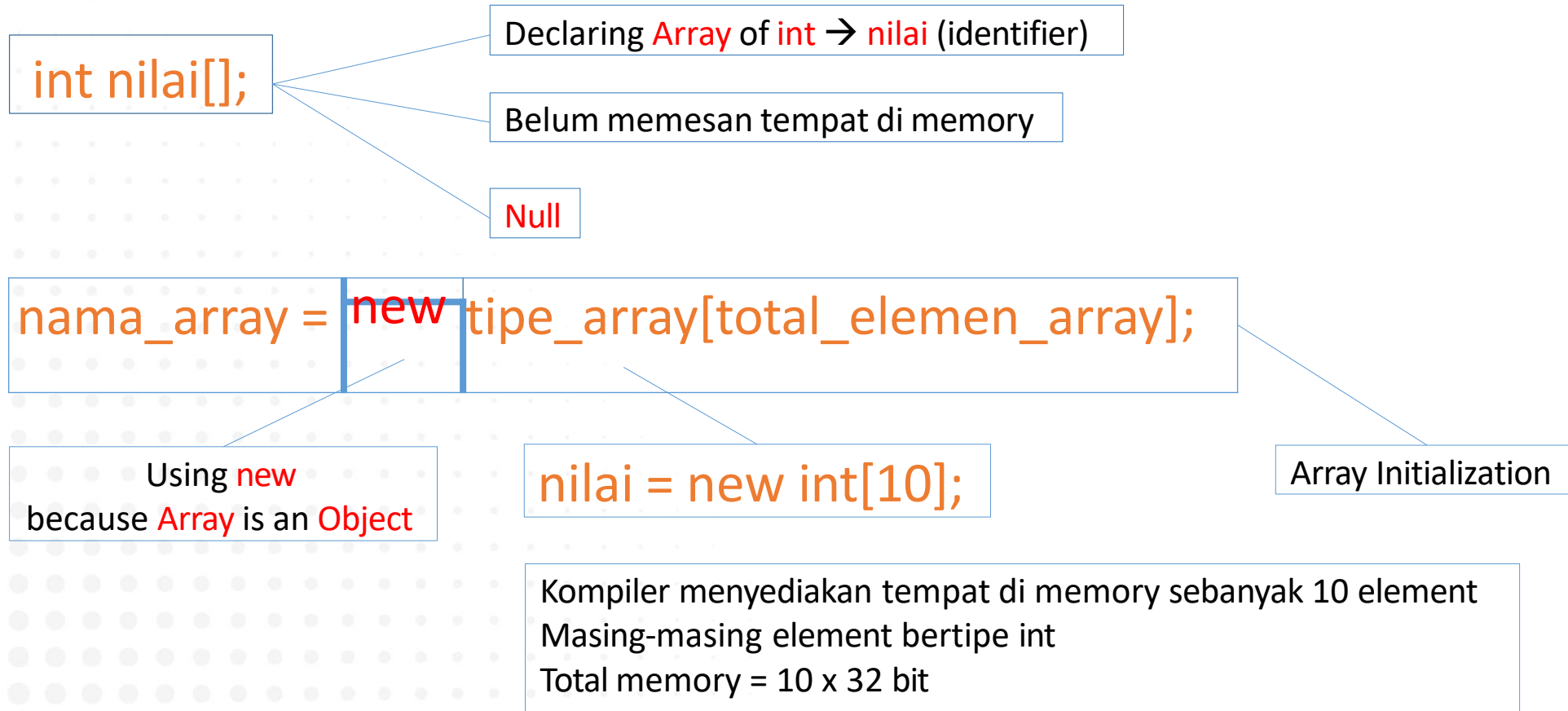
- *blank space* tidak berpengaruh terhadap peletakan tanda [].
- Contoh:

```
tipe_array nama_array [];
```

```
tipe_array [] nama_array;
```

```
tipe_array []nama_array;
```

Initializing Array



Assigning Array

- Array harus diisi dengan nilai yang sesuai dengan tipe array-nya.
- Beritahukan kepada kompiler Java elemen manakah dari array yang akan diisi.
- Format :

```
nama_array[elemen_array] = nilai;
```

- Contoh :

```
int nilai[];
```

```
nilai = new int[2];
```

```
nilai[1] = 100;
```

- Di baris ketiga, kita berusaha mengakses elemen array pada indeks ke-1 dan memberinya dengan nilai 100.

Declaring and Initializing Array

- Dimungkinkan melakukan deklarasi dan inisialisasi array hanya pada satu baris *statement*.
- Format:

```
tipe_array nama_array[] = new tipe_array[total_elemen_array];
```

- Contoh :

```
int nilai[] = new int[5];
```

Declaring, Initialization, Assigning Array

- Format

```
tipe_array nama_array[] = {nilai_indeks_0, nilai_indeks_1, ... , nilai_indeks_n};
```

- Contoh :

```
int nilai[] = {70, 65, 85};
```

Array multi dimensi

- Multidimensional Array
 - adalah Array dimana setiap element nya juga berupa array

Length is 5

Length is 10

Declaring Array Multi Dimentions

<code>int nilai[];</code>	→ berdimensi 1
<code>char huruf[][];</code>	→ berdimensi 2
<code>double total[][][];</code>	→ berdimensi 3
<code>String[][] nama;</code>	→ berdimensi 2
<code>float[] panjang[];</code>	→ berdimensi 2
<code>short[][] tinggi[];</code>	→ berdimensi 3

Initializing Array Multi Dimention

```
int x[ ][ ] = new int[3][4];
```

berdimensi 2, dengan 3 elemen di dimensi ke-1 dan 4 elemen di dimensi ke-2

```
double[ ][ ][ ] balok = new double[3][4][2];
```

berdimensi 3, dengan 3 elemen di dimensi ke-1, 4 elemen di dimensi ke-2 dan 2 elemen di dimensi ke-3

Array multidimensions tak beraturan

- Diperkenankan untuk membuat suatu array yang berdimensi tidak beraturan.
- Contoh :

```
int x[ ][ ] = new int[3][ ];
```

```
x[0] = new int[2];
```

```
x[1] = new int[3];
```

```
x[2] = new int[1];
```

Mengetahui total elemen array

- Untuk mengetahui panjang dari suatu array yang telah dibuat, dapat memakai properti *length*.
- Contoh :

```
int x[] = new int[3];  
x.length    → 3
```

```
int y[][] = new int[2];  
y[0] = new int[3];  
y[1] = new int[5];  
y[0].length → 3  
y[1].length → 5
```

Merubah total elemen array ??

- Tidak diperkenankan merubah total elemen array.
- Diperkenankan membuat array baru dengan nama object yang sama
(Tetapi total elemen array yang sebelumnya akan tergantikan dengan yang baru.)

```
byte posisi[] = new byte[2];
```

```
posisi[0] = 55;
```

```
posisi[1] = 68;
```

```
byte posisi[] = new byte[5];
```

- Yang terjadi : total elemen array yang sebelumnya dan juga nilai dari setiap elemennya akan hilang dan tergantikan dengan array yang baru.

Mengkopi elemen array

- Isi dari suatu array dapat di kopi pada array yang lain dengan memanfaatkan method `arraycopy()` pada class `System`.
- Format :

`System.arraycopy(array1, p1, array2, p2, n);`

dimana :

`array1` = array asal/sumber pengkopian

`array2` = array tujuan pengkopian

`p1` = posisi indeks awal pengkopian pada array asal

`p2` = posisi indeks awal pengkopian pada array tujuan

`n` = banyaknya elemen array yang akan dikopi

Mengkopi element array

- Contoh :

```
int lama[ ] = {1, 2, 3, 4, 5, 6};
```

```
int baru[ ] = {10, 9, 8, 7, 6, 5, 4, 3, 2, 1};
```

```
System.arraycopy(lama, 2, baru, 5, 3);
```

- Elemen-elemen array baru akan bernilai masing 10, 9, 8, 7, 6, 3, 4, 5, 2, 1

Referensi array

- Suatu array juga dapat me-refer (merujuk) ke array yang lain, dengan kata lain merujuk pada alamat memori yang sama.
- Contoh :
 - `int nilai[] = {10, 20, 30};`
 - `int result[];`
 - `result = nilai;`
- Di baris ketiga, assign array result ke array nilai.
- Akibatnya, array result akan me-refer (merujuk) pada array nilai, sehingga kedua array tersebut merujuk alamat memori yang sama.
 - Pada saat nilai elemen pada array result diubah, misalnya dengan :
`result[1] = 50;`
- maka nilai dari indeks ke-1 pada array nilai (`nilai[1]`) juga ikut berubah menjadi 50.



Tugas

1. Apakah yang dimaksud dengan array?
2. Buatlah contoh mendeklarasikan, memberi nilai dan mengakses elemen array!
3. Bagaimana cara mengetahui panjang array?
4. Uraikan pengetahuan anda mengenai array multidimensi!
5. Bagaimana cara mengkopi isi array?
6. Apakah yang dimaksud dengan referensi array. Beri contoh!

1. Oracle Java Documentation, The Java™ Tutorials, <https://docs.oracle.com/javase/tutorial/>, Copyright © 1995, Oracle 2015.
2. Tita Karlita, Yuliana Setrowati, Rizky Yuniar Hakkun, Pemrograman Berorientasi Obyek, PENS-2012
3. Sun Java Programming, Sun Educational Services, Student Guide, Sun Microsystems, 2001.
4. John R. Hubbard, Programming With Java, McGraw-Hill, ISBN: 0-07-142040-1, 2004.
5. Patrick Niemeyer, Jonathan Knudsen, Learning Java, O'reilly, CA, ISBN: 1565927184, 2000.
6. Philip Heller, Simon Roberts, Complete Java 2 Certification Study Guide, Third Edition, Sybex, San Francisco, London, ISBN: 0-7821-4419-5, 2002.
7. Herbert Schildt, The Complete Reference, Java™ Seventh Edition, Mc Graw Hill, Osborne, ISBN: 978-0-07-163177-8, 2007