

Array (Larik)



Pendahuluan

- Array biasanya digunakan untuk mengelompokkan data atau obyek yang memiliki tipe yang sama
- Array memungkinkan untuk mengacu ke sekumpulan obyek dengan nama yang sama
- Array dapat dideklarasikan dengan tipe apa saja, baik itu yang bertipe data primitif maupun obyek

Apa itu Array?

- **Array** merupakan objek yang digunakan untuk menyimpan banyak data (data/variabel bisa berupa tipe data primitif atau objek) dengan tipe yang sama.
- Data-data pada array disimpan dalam urutan tertentu yang dinamakan indeks
- Sehingga untuk merujuk pada lokasi tertentu atau element array tertentu maka kita menentukan nama array dan posisi elemen array yg diidentifikasi menggunakan indeks.

Apa itu Array?

- Setiap nomor **indeks dalam array dialokasikan dalam ruang memori** tertentu, sehingga pengguna harus menghindari pernyataan array yang ukurannya lebih besar daripada yang dibutuhkan.
- Terdapat tiga langkah untuk membuat Array
 - Mendeklarasikan variabel array
 - MenCreate array beserta ukurannya
 - Memberikan sebuah nilai pada setiap elemen array

Pendeklarasian Array

- Contoh pendeklarasian Array sbb :
 - `int[] diskon; atau int diskon[]`
 - `double[] harga;`
 - `Dimension[] dimensi;`
 - `float[][] matrikA;`
- Deklarasi yang tidak legal
 - `int[5] diskon;`

Create Array

- Dalam bahasa pemrograman Java, **Array** merupakan sebuah obyek meskipun ia terdiri dari elemen yang bertipe data primitif
- Seperti halnya kelas yang lain, ketika mendeklarasikan Array belum dibentuk sebuah obyek Array
- Deklarasi Array hanya membuat sebuah referensi yang dapat digunakan untuk mengacu ke sebuah obyek Array.

Create Array

- Seperti halnya dalam pembuatan obyek, Array dibuat dengan menggunakan keyword *new*

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

Atau

```
double[] harga = new double[5];
```

Create Array

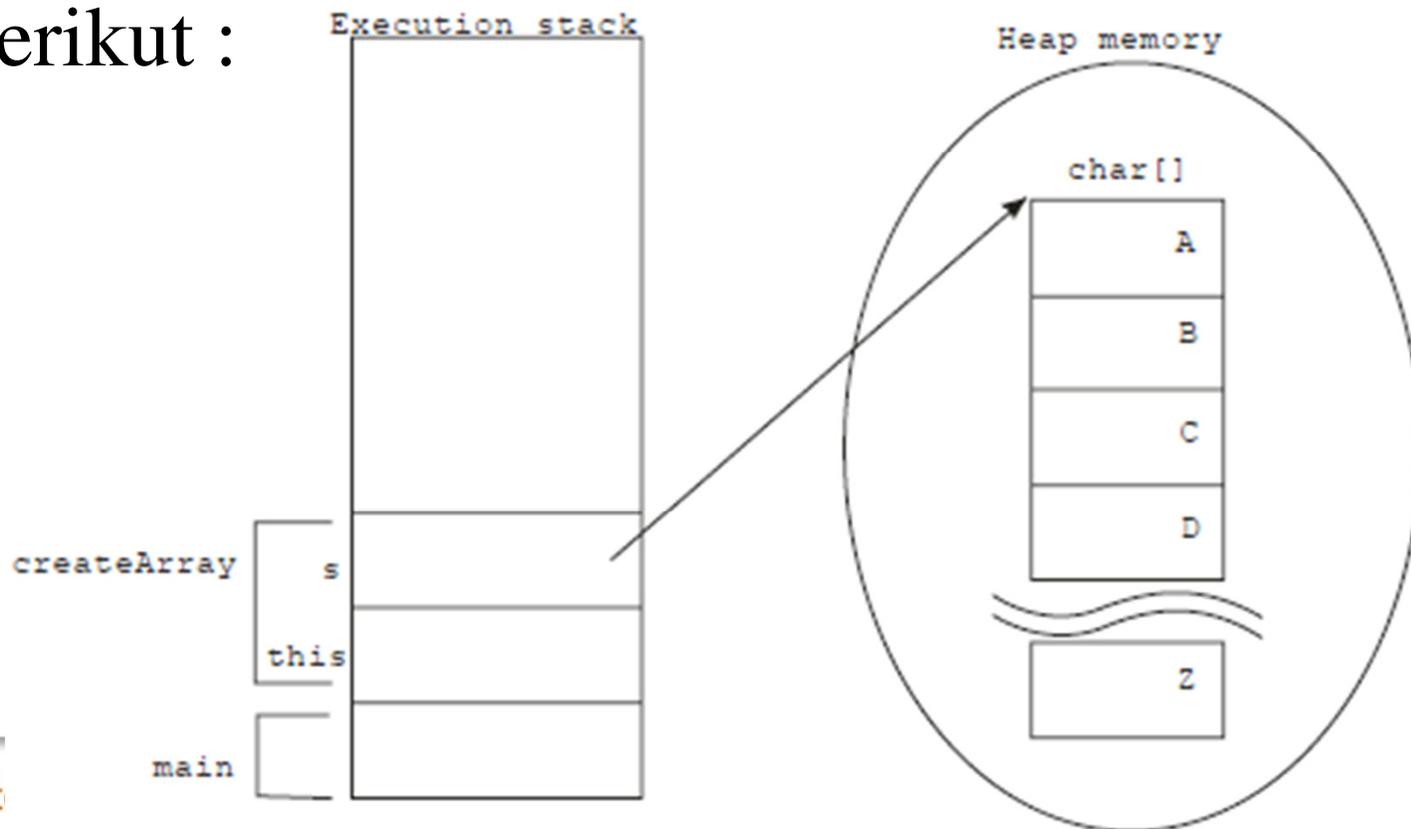
- Ketika kita membuat array dengan memberikan nilai indeks 5 pada definisi, maka array tersebut dapat menyimpan 5 elemen dimulai dari index ke 0 hingga 4.

```
double[] harga = new double[5];
```

Perhatikan Contoh Berikut

```
char[] createArray {  
    char[] s;  
    s = new char[26];  
    for(int i=0; i<26; i++) {  
        s[i] = (char) ('A' + i);  
    }  
    return s;  
}
```

- Penggalan kode tersebut mempunyai representasi dalam heap memory sebagai berikut :



Memberikan Nilai Pada Array

- Ketika kita memiliki variabel bertipe array maka kita dapat **merujuk pada elemen tertentu pada array dengan menentukan nama variabel dan indeks** yg mewakili elemen dari array tersebut.
- Dalam array tidak dibenarkan merujuk pada elemen array yg berada di luar batas maksimal indeks array, atau diluar ukuran array.

Memberikan Nilai Pada Array

- Perhatikan contoh berikut :
- Isi dari array : nilai primitif

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

```
scores[0] = 75;
```

```
scores[1] = 80;
```

```
scores[2] = 100;
```

```
scores[3] = 50;
```

```
scores[4] = 60;
```

Dengan nilai 5 maka array memiliki 5 elemen array, maka indeks awal array dimulai dari 0 hingga indeks ke 4.

Inisialisasi Array

- Ketika membuat sebuah Array, setiap elemen di dalamnya akan diinisialisasi dengan nilai *default*.
- Sebagai contoh ketika array didefinisikan sebagai array dari tipe *integer* maka nilai defaultnya adalah 0, jika array didefinisikan sebagai array dari tipe *String* maka nilai defaultnya adalah null

Inisialisasi Array

- Java memiliki cara singkat untuk inisialisasi array sbb :

```
String names[] =  
    {"Georgianna", "Jen", "Simon"};
```

Inisialisasi Array

```
String names[] =  
    {"Georgianna", "Jen", "Simon"};
```

- Kode di atas adalah sama dengan kode berikut ini:

```
String names[];  
names[0] = "Georgianna";  
names[1] = "Jen";  
names[2] = "Simon";
```

What???

Ada Pertanyaan ?

Why ???

Praktikum 5.1 : Array

Buatlah kelas baru dengan nama arrayKota

```
public class arrayKota {  
  
    void Tampil()  
    {  
        String[] kota;  
        kota=new String[5];  
        kota[0]="Jakarta";  
        kota[1]="Bandung";  
        kota[2]="Semarang";  
        kota[3]="Surabaya";  
        kota[4]="Denpasar";  
  
        System.out.println("Tampilkan kota ke 2 "+kota[2]);  
        System.out.println("-----");  
  
        for(int i=0;i<kota.length;i++)  
        {  
            System.out.println("Index ke "+i+" berisi :"+kota[i]);  
        }  
  
        kota[2]=kota[3];  
        kota[3]=kota[4];  
  
        System.out.println("-----");  
  
        for(int i=0;i<kota.length;i++)  
        {  
            System.out.println("Index ke"+i+" berisi :"+kota[i]);  
        }  
    }  
}
```

```
public class arrayKota_Main {  
    public static void main(String[] args)  
    {  
        arrayKota kota = new arrayKota();  
        kota.Tampil();  
    }  
}
```

Pada kelas
arrayKota_Main
tambahkan main program

Array Multidimensi

- Dalam Java, dapat dibuat Array dari Array sehingga dinamakan Array multidimensi.
- Array dapat kita buat dalam bentuk satu dimensi atau lebih dari satu dimensi, **Array multidimensi merujuk pada konsep array dengan dimensi lebih dari satu dimensi.**
- Penggunaan umum array multidimensi merupakan tabel yang berisi informasi yang diatur dalam baris dan kolom.

Array Multidimensi

- Untuk mengidentifikasi elemen array tersebut maka kita harus menambahkan
 - Indeks yg menyatakan elemen baris dan
 - Indeks kedua yg menyatakan elemen kolom
- Array yg menggunakan dua indeks untuk mengidentifikasi element tertentu disebut array dua dimensi
- Sebagai contoh **Array dengan dimensi satu disebut *vektor***, dan **array berdimensi dua disebut *matrik***.

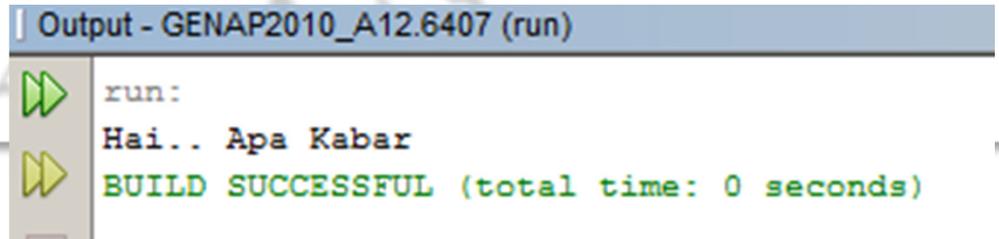
Contoh Array dengan 2 Dimensi

- Ketika kita membuat definisi array dua dimensi maka kita menyertakan 2 indeks pada definisi array.
- Indeks yg pertama menyatakan baris dan yg ke dua menyatakan kolom

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

Contoh Array Dua Dimensi

```
public static void main(String[] args)
{
    String[][] kata=new String[2][2];
    kata[0][0]="Halo";
    kata[0][1]="Apa Kabar";
    kata[1][0]="Selamat Petang";
    kata[1][1]="Hai..";
    System.out.println(kata[1][1] +
        " "+kata[0][1]);
}
```



```
Output - GENAP2010_A12.6407 (run)
run:
Hai.. Apa Kabar
BUILD SUCCESSFUL (total time: 0 seconds)
```

Array Multidimensi Non-Rectangular

- Beberapa cara kita mendefinisikan **array multidimensi secara rectangular** artinya ukuran **dimensi array semuanya sama** yakni ukuran indeks pada baris dan kolom sama.
- Namun dalam java kita juga dapat membuat array dengan ukuran indeks baris dan kolom yg tidak sama atau Non-Rectangular

Array Multidimensi Non-Rectangular

- Perhatikan definisi berikut :

```
int twoDim [][] = new int[4][];
```

- `twoDim[0] = new int[2];`
- `twoDim[1] = new int[4];`
- `twoDim[2] = new int[6];`
- `twoDim[3] = new int[8];`

Batasan Array

- Dalam bahasa Java, **index Array dimulai dengan angka nol**. Jumlah elemen di dalam Array disimpan sebagai bagian dari obyek Array
- Untuk mendapatkan batasan array maka kita dapat memanggil atribut *length* pada objek array tersebut.

Batasan Array

```
int list[] = new int[10];  
for(int i = 0; i < list.length; i++) {  
System.out.println(list[i]);  
}
```

- Dengan menggunakan atribut *length*, pemeliharaan kode program menjadi mudah karena program tidak perlu mengetahui jumlah elemen Array pada saat kompilasi

Array Resizing

- Setelah membuat obyek Array, ukuran Array tersebut tidak dapat diubah.
- namun demikian, dapat digunakan variabel referensi yang sama untuk menunjuk ke sebuah obyek Array baru

```
int myArray[] = new int[6];  
myArray = new int[10];
```

- Ketika kita merubah ukuran array sebenarnya kita membuat objek baru dengan menggunakan pernyataan *new* maka informasi dari objek sebelumnya akan hilang.

Menyalin Array

- Bahasa Java menyediakan method khusus untuk menyalin Array yaitu dengan menggunakan method *arrayCopy()* dari kelas `System`
- Contoh :

```
// array semula
int myArray[] = {1, 2, 3, 4, 5, 6};
// array engan elemen yang lebih banyak
int hold[] = {10, 9, 8, 7, 6, 5, 4, 3, 2, 1};
//menyalin semua elemen myArray ke hold
// dimulai dengan indeks ke 0
System.arraycopy(myArray, 0, hold, 0, myArray.length);
```

Perhatikan Contoh Berikut

```
public static void main(String[] aksi)
{

    String[][] kata=new String[2][2];
    kata[0][0]="Halo";
    kata[0][1]="Apa Kabar";
    kata[1][0]="Selamat Petang";
    kata[1][1]="Hai..";

    String[][] kalimat = new String[2][2];
    System.arraycopy(kata, 0, kalimat, 0, kata.length);
    System.out.println(kata[1][1] + " " +kalimat[1][0]);

}
```

Output - GENAP2010_A12.6407 (run)

```
run:
Hai.. Selamat Petang
BUILD SUCCESSFUL (total time: 0 seconds)
```

What???

Ada Pertanyaan ?

Why ???

Praktikum 5.2 : Array Multidimensi

Buatlah kelas baru dengan
nama arrayNegara

```
public class arrayNegara {  
  
    void Display()  
    {  
  
        String negara[][] = {"Indonesia", "Malaysia"},  
                               {"Jepang", "Korea", "Filipina"},  
                               {"Amerika"}};  
  
        System.out.println("Negara pada baris ke 0 kolom ke 1 adalah " + negara[0][1]);  
        System.out.println("Negara pada baris ke 1 kolom ke 2 adalah " + negara[1][2]);  
        System.out.println("Negara pada baris ke 2 kolom ke 0 adalah " + negara[2][0]);  
  
    }  
    public static void main(String[] args)  
    {  
        arrayNegara negara = new arrayNegara();  
        negara.Display();  
    }  
}
```

Terima kasih



Daftar Pustaka

- Java™ Tutorial, Third Edition: A Short Course on the Basics, Addison Wesley , 2000.
- Kathy Sierra & Bert Bates, “Sun Certified Programmer for Java tm 6 Study Guide”, McGraw-Hill Companies, 2008.
- Liem, I. (2003). Diktat Kuliah Pemrograman Berorientasi Objek. Departemen Teknik Informatika Institut Teknologi Bandung