

PERTEMUAN 17

ARRAY / LARIK

A. TUJUAN PEMBELAJARAN

1. Mahasiswa memahami konsep dari Array
2. Mahasiswa dapat membedakan jenis array satu dimensi dan dua dimensi
3. Mahasiswa dapat mengerti fitur pada array dan mendeklarasikan array
4. Mahasiswa dapat mempraktekkan penggunaan *array* satu dimensi dan multi dimensi

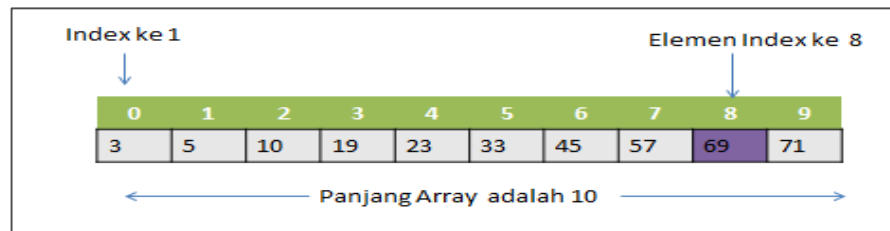
B. URAIAN MATERI

Array adalah objek yang menyimpan banyak variabel dengan tipe data yang sama. Data dapat menampung tipe data primitif serta *reference* objek. Array pada Java disimpan menggunakan indeks, elemen pertama array disimpan pada indeks ke-0, elemen ke-2 disimpan pada indeks ke-1 dan seterusnya.

Sebenarnya sebagian besar tipe koleksi di Java yang merupakan bagian dari paket `java.util` menggunakan array secara internal dalam fungsinya. Karena Array adalah objek, mereka dibuat selama runtime. Panjang array adalah tetap.

Fitur Array

1. Array adalah objek
2. Array bahkan dapat menyimpan variabel *reference* dari objek lain
3. Array dibuat selama runtime
4. Array dinamis, dibuat di heap
5. Panjang Array adalah tetap



Gambar 17. 1 1 Ilustrasi Objek array

1. Jenis jenis Array

Berdasarkan jenis *Array* dapat dibedakan terdiri dari :

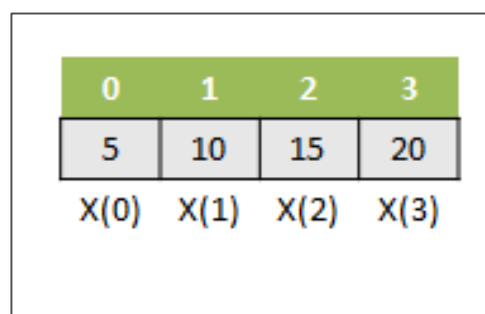
- Array satu dimensi
- Array dua dimensi

Array satu dimensi (atau array dimensi tunggal) adalah jenis array linier. Untuk mengakses elemen-elemennya melibatkan satu subskrip dapat mewakili indeks baris atau kolom.

- Array dapat berupa variabel atau berupa konstanta.
- Array dapat berupa array dari tipe data primitif int, long, float, String atau bahkan objek
- Definisi array 1 dimensi misalnya :

`int [] X = { 5,10,15,20 } ;`

adalah mendefinisikan array 1 dimensi dari int yang berelemen data 5,10,15,20, ilustrasi gambar array nya adalah sebagai berikut :



Gambar 17. 2 Definisi Array 1 Dimensi

2. Deklarasi Array satu dimensi :

Untuk mendeklarasikan Variabel array seperti kita mendeklarasikan variabel dengan tipe yang diinginkan, hanya untuk deklarasi variabel *array*

menambahkan [] setelah menentukan tipe datanya. Berikut adalah contoh deklarasi array Java sederhana:

```
int[] NamaArray;  
  
String[] stringArray;  
  
MyClass[] myClassArray;  
  
int    intArray[];  
  
String    stringArray[];  
  
MyClass    myClassArray[];
```

Pada Umumnya deklarasi array menempatkan tanda kurung siku [] setelah tipe data (mis. String []) ditentukan dan bukan setelah nama variabel. Karena lebih mudah untuk membaca kode ketika tanda kurung siku ditempatkan tepat setelah tipe data dalam deklarasi array.

3. Instansiasi / Membuat Object dan mengakses Array

Saat kita mendeklarasikan variabel array, kita hanya mendeklarasikan variabel (reference) ke array itu sendiri. Deklarasi sebenarnya tidak membuat array. Untuk membuat array penulisan nya seperti dibawah ini:

```
int[] NamaArray;  
  
NamaArray = new int[10];
```

Contoh diatas adalah membuat array dengan nama `NamaArray` bertipe data primitif `int` dengan ruang atau jumlah elemen 10 variabel di dalamnya.

Kita juga dapat membuat larik reference objek String. seperti dibawah ini :

```
String[] stringNamaArray = new String[10];
```

Jika kita ingin memberikan nilai variabel yang akan dimasukkan ke dalam array, kita dapat menggunakan literal array. Berikut adalah bagaimana tampilan literal array dalam kode Java:

```
int[] angka= { 1,2,3,4,5,6,7,8,9,10 };  
  
String[] huruf= {"satu", "dua", "tiga"};
```

Contoh sederhana program *array* pada java

KotaArray.java

```
public class KotaArray{  
    public static void main(String[] args){  
        String[] kota; //deklarasi variabel array  
  
        kota = new String[4]; // membuat objek  
        array  
  
        // mengisi elemen array  
        kota[0] = " Bandung";  
        kota[1] = " Semarang ";  
        kota[2] = " Surabaya ";  
        kota[3] = " Medan";  
  
        // menampilkan elemen array  
        System.out.println(kota[0]);  
        System.out.println(kota[1]);  
        System.out.println(kota[2]);  
        System.out.println(kota[3]);  
  
    }  
}
```

Output :

```
Bandung  
Semarang  
Surabaya
```

Medan

Contoh sederhana program dengan memberikan nilai literal *array* pada java

```
public class ArrayKota2{  
    public static void main(String[] args){  
        String[]kota={"Bandung","Semarang","Surabaya","Medan"  
" };  
  
        // menampilkan elemen array  
        System.out.println(kota[0]);  
        System.out.println(kota[3]);  
        System.out.println(kota[2]);  
        System.out.println(kota[1]);  
    }  
}
```

Output

Bandung
Medan
Surabaya
Semarang

Jumlah Elemen *array* dapat diketahui menggunakan variable *instance length*

ArraykotaLength.java

```
public class arraykotalength {  
    public static void main(String[] args) {  
        // TODO code application logic here  
String[] kota = {"Medan","Jakarta", "Surabaya",  
"Semarang"};
```

```
// menampilkan elemen array

for(int i=0; i<kota.length; i++)

    System.out.println(kota[i]);

}

}
```

Output :

```
Medan
Jakarta
Surabaya
Semarang
```

Contoh sederhana program *array* dengan type data Integer

ArrayInt.java

```
class ArrayInt{

    public static void main(String[] args) {

        // declares an array of integers

        int[] intArray;

        // alokasi memori untuk 10 element type
integer

        intArray = new int[10];

        // initialize elemen pertama

        intArray[0] = 100;

        // initialize elemen Kedua

        intArray[1] = 200;

        // dan seterusnya
```

```
intArray[2] = 300;
intArray[3] = 400;
intArray[4] = 500;
intArray[5] = 600;
anArray[6] = 700;
intArray[7] = 800;
intArray[8] = 900;
intArray[9] = 1000;

System.out.println("Element index 0: "
    + intArray[0]);
System.out.println("Element index 1: "
    + intArray[1]);
System.out.println("Element index 2: "
    + intArray[2]);
System.out.println("Element index 3: "
    + intArray[3]);
System.out.println("Element index 4: "
    + intArray[4]);
System.out.println("Element index 5: "
    + intArray[5]);
System.out.println("Element index 6: "
    + intArray[6]);
System.out.println("Element index 7: "
    + intArray[7]);
System.out.println("Element index 8: "
    + intArray[8]);
```

```
        System.out.println("Element index 9: "
                            + intArray[ 9]);
    }
}
```

Output :

```
Element index 0: 100
Element index 1: 200
Element index 2: 300
Element index 3: 400
Element index 4: 500
Element index 5: 600
Element index 6: 700
Element index 7: 800
Element index 8: 900
Element index 9: 1000
```

4. Menyalin Array (Copying)

Class Sistem memiliki method *arraycopy* yang dapat kita gunakan untuk menyalin data secara efisien dari satu array ke array lain

Bentuk umum penulisan *copying array*

```
public static void arraycopy(Object src, int srcPos,
                             Object dest, int destPos, int length)
```

Dua argumen Objek menentukan array sumber yang akan disalin dan array sebagai target tujuan. Tiga argumen int menentukan posisi awal dalam array sumber, posisi awal dalam array tujuan, dan jumlah elemen array yang akan disalin.

Program berikut, mendeklarasikan array elemen String. Array ini menggunakan metode `System.arraycopy` untuk menyalin urutan komponen array ke dalam array kedua:

Arraycopy.java

```
class ArrayCopy{

    public static void main(String[] args) {

        String[] copyFrom = {

            "Corretto", "Cortado", "Doppio",
            "Affogato", "Americano", "Cappuccino", "Espresso",
            "Frappucino", "Freddo", "Lungo", "Macchiato", "Marocchino",
            "Ristretto" };

        String[] copyTo = new String[7];

        System.arraycopy(copyFrom, 2, copyTo, 0, 7);

        for (String coffee : copyTo) {

            System.out.print(coffee + " ");

        }

    }

}
```

Output :

```
Doppio Affogato Americano Cappuccino Espresso Frappucino
Freddo
```

5. Manipulasi Array

Array adalah merupakan konsep yang *powerfull* dan berguna dalam pemrograman. Java SE menyediakan metode untuk melakukan beberapa manipulasi yang paling umum terkait dengan *array*. Misalnya pada, contoh *ArrayCopy* menggunakan metode *arraycopy* dari kelas *System* alih-alih melakukan iterasi secara manual melalui elemen *array* sumber dan menempatkan masing-masing elemen ke dalam *array* tujuan. Ini dilakukan di belakang layar, sehingga memungkinkan pengembang hanya menggunakan hanya satu baris kode untuk memanggil metode.

Java SE menyediakan beberapa metode untuk melakukan manipulasi array (tugas umum, seperti menyalin, menyortir, dan mencari array) di kelas `java.util.Arrays`. Misalnya, contoh sebelumnya dapat dimodifikasi untuk menggunakan metode `copyOfRange` dari kelas `java.util.Arrays`, seperti yang Anda lihat dalam contoh `ArrayCopy`. Perbedaannya adalah menggunakan metode `copyOfRange` tidak mengharuskan kita membuat *array* tujuan sebelum memanggil metode tersebut, karena *array* tujuan dikembalikan oleh metode:

```
class ArrayCopyOfDemo {  
    public static void main(String[] args) {  
        String[] copyFrom = {  
            "Affogato",    "Americano",    "Cappuccino",    "Corretto",  
            "Cortado",     "Doppio",     "Espresso",     "Frappuccino",  
            "Freddo",    "Lungo",    "Macchiato",    "Marocchino",  
            "Ristretto" };  
  
        String[] copyTo=java.util.Arrays.copyOfRange(copyFrom,  
        2, 9);  
  
        for (String coffee : copyTo) {  
            System.out.print(coffee + " ");  
        }  
    }  
}
```

6. Array multidimensi

Array multidimensi atau *array* dua dimensi pada Java direpresentasikan sebagai array dari array satu dimensi dengan tipe data yang sama. Penggunaan Array multidimensi sebagian besar digunakan untuk mewakili nilai dari tabel dimana terdapat nilai baris dan kolom.

Array multidimensi adalah array dari array , dengan konsep pengaksesan `[noBaris][noKolom]`

Deklarasi array multidimensi :

```
dataType[][] arrayRefVariable; (or) dataType arrayRefVariable[][];
```

Pendeklarasian Terdapat 2 kurung buka dan kurung tutup bracket berpasangan setelah menentukan tipe datanya :

Contoh :

```
int[][] VarArray=new int[3][3]; //3 baris dan 3 column
```

```
int[][] VarArray= {{10, 20, 30},{11, 21, 31},{12, 22, 32}} }
```

Singkatnya, array dua dimensi berisi elemen array satu dimensi. Ini diwakili oleh dua indeks di mana indeks pertama menunjukkan posisi array baris dan indeks kedua mewakili posisi elemen kolom dalam array tersebut.

	Kolom 0	Kolom 1	Kolom 2
Baris 0	10	20	30
Baris 1	11	21	31
Baris 2	12	22	32

Gambar 17. 3 Array multi dimensi



Gambar 17. 4 Meng akses Array multi dimensi

VarArray [1][2] akan membrikan nilai **31**

7. Contoh program sederhana Array multidimensi

Contoh 1

```
public class Main
{
    public static void main(String[] args) {
        //2-d array initialised with values
        int[][] intArray = { { 1, 2 }, { 3, 4 }, {5,6}};
        //print the array
        System.out.println("Inisialisasi array multi
dimensi :");
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 2; j++) {
                System.out.print(intArray [i][j] + " ");
            }
            System.out.println();
        }
    }
}
```

Output

```
Inisialisasi array multi dimensi

1    2
3    4
5    6
```

Contoh 2

```
class ArrayMultidimensi {  
    public static void main(String[] args) {  
        // TODO code application logic here  
        String[][] nama = {"Mas ", "Pak ", "Bu " ,  
                            "Mbak "}, {"Agus ", "Iyus ", "Nabila "};  
        System.out.println(nama[0][0] + nama[1][0]);  
        System.out.println(nama[0][1] + nama[1][1]);  
        System.out.println(nama[0][2] + nama[1][2]);  
  
    }  
}
```

Output :

Mas Agus

Pak Iyus

Bu Nabila

C. LATIHAN /TUGAS

1. Modifikasi program Arraylength.java diatas , gunakan class scanner untuk menerima inputan dari keyboard

Contoh hasil run

```
Masukan Jumlah Kota : 4  
Kota ke 1 : Jakarta  
Kota ke 2 : Bandung  
Kota ke 3 : Surabaya  
Kota ke 4 : Makasar  
Kota-kota yang dimasukan  
Jakarta
```

```
Bandung  
Surabaya  
Makasar
```

2. Buat array multidimensi untuk nama negara dan ibukotanya, Akses array dan tampilkan di layar outputnya adalah sebagai berikut :

```
Ibukota Indonesia adalah Jakarta  
Ibukota Jepang adalah Tokyo  
Ibukota Iran adalah Teheran  
Ibukota Malaysia adalah Kuala Lumpur  
Ibukota Thailand adalah Bangkok
```

D. REFERENSI

- Horstmann Cay S., (2011). *Big Java 4th Edition*, san jose university , united state Of America. RRD jefferson city publishing.
- Deitel Paul , Deitel Harvey, (2012) Java how to program eighth edition, pearson education, Boston Massachusetts , USA, publishing as prentice hall.
- Rose Cristhoper, (2017), Java Succinctly Part 2, Morrisville, NC 27560, USA, Syncfusion, Inc.
- Downey Allen B. , Mayfield Chris, (2017), Think Java, Needham, Massachusetts, USA, Green Tea Press
- Hayes Helen, (2021), BeginnersBook.com, <https://beginnersbook.com/java-tutorial-for-beginners-with-examples/>, di akses pada tanggal 21 November 2021.
- Sonoo Jaiswal , (2021) JavaTpoint offers college campus training , <https://www.javatpoint.com/java-tutorial>, diakses pada tanggal 1 Desember 2021
- Wahono Satrio Romi, Java Fundamental, <https://romisatriawahono.net/java/>, diakses pada tanggal 5 Desember 2022