Michael Jonathan Setiawan

58170468

Pertemuan 11 - *Array* dan *Array List*

*Array*

*Array* adalah sebuah variabel yang menyimpan sekumpulan data yang mempunyai tipe yang sama atau kumpulan dari nilai-nilai data yang bertipe sama (misal *integer*). Setiap data menempati alamat memori atau lokasi yang berbeda dan selanjutnya disebut dengan elemen *array* yang dapat diakses melalui indeks yang terdapat di dalamnya.

Indeks *array* dimulai dari 0 (jika indeks 0-9, maka panjang *array* adalah 10). Elemen yang disimpan pada *array* dapat berupa tipe primitif (*int*, *float*, *etc*) atau objek (instan dari *class*).

1. *Array* Satu Dimensi

*Array* satu dimensi merupakan *array* yang terdiri dari n buah kolom atau terdiri dari 1 subskrip *array* saja. Setiap elemen dapat diakses melalui indeks yang terdapat di dalamnya.

Deklarasi:

TipeData[] nama\_var\_array = new TipeData[Ukuran];

TipeData : *declare* *type data* yang digunakan.

Ukuran : jumlah maksimum elemen *array*.

Menciptakan Objek *Array*

Deklarasi:

namaVariabel = new tipePrimitif[jumlahElemen];

namaVariabel = new namaKelas[jumlahElemen];

Contoh:

nomor = new int[7];

kota = new String[8];

Bentuk singkat deklarasi variabel dan objek *array*:

int [] nomor = new int[7];

String[] kota = new String[8];

Mengakses *Array* satu dimensi

Suatu *array*, dapat diakses dengan menggunakan *subscript* atau *index*-nya.

Deklarasi:

namaVariabelArray[nomorElemen];

Contoh:

kota[0] = “Surabaya”;

1. *Array* Dua Dimensi

*Array* dua dimensi merupakan *array* yang terdiri dari n buah baris dan m buah kolom, bentuknya dapat digambarkan seperti matriks atau tabel. Indeks pertama baris dan indeks kedua kolom. Biasanya digunakan untuk pendataan nilai, pendataan penjualan, dan lain sebagainya.

Deklarasi:

TipeData[][] nama\_var\_array = new TipeData[Ukuran\_baris][Ukuran\_kolom];

TipeData : *declare* *type data* yang digunakan.

Ukuran\_baris : jumlah maksimum elemen baris dalam *array*.

Ukuran\_kolom : jumlah maksimum elemen kolom dalam *array*.

Inisialisasi *Array* Dua Dimensi

Bentuk umum pengaksesan:

TipeData[][] nama\_var\_array = { nilai array };

Contoh:

int[][] nil\_akhir = { {150, 159, 230}, {100, 125, 150}, {210, 125, 156} };

1. *Array* Tiga Dimensi

*Array* dua dimensi tersusun dalam bentuk baris, kolom, dan isi dari baris. Indeks pertama menunjukkan baris, indeks kedua menunjukkan kolom, dan indeks ketiga menunjukkan isi dari baris.

Deklarasi:

TipeData[][][] nama\_var\_array = new TipeData[Ukuran\_baris][Jumlah\_isi][Ukuran\_kolom];

TipeData : *declare* *type data* yang digunakan.

Ukuran\_baris : jumlah maksimum elemen baris dalam *array*.

Jumlah\_isi : jumlah isi dari baris.

Ukuran\_kolom : jumlah maksimum elemen kolom dalam *array*.

Inisialisasi *Array* Dua Dimensi

Suatu *array* dapat diakses dengan menggunakan *subscript* atau *index*-nya.

Bentuk umum pengaksesan:

TipeData[][][] nama\_var\_array = { nilai array };

1. *Array List*

*Array List class* mengelola urutan *object* yang dapat bertambah dan berkurang sesuai dengan keperluan. *Array List class* menyediakan banyak *method* untuk berbagi keperluan misal menambah atau menghapus elemen. *Array List* merupakan suatu *collection* pada java *library* yang termasuk pada Java Util. *Array List* sering digunakan oleh para *developer* untuk menyimpan dan memetakan data.

Sifatnya sangat fleksibel dan sangat mudah digunakan dalam berbagai kasus untuk penyimpanan data yang tidak tentu jumlahnya dan juga umumnya hanya bisa menyimpan *value* dengan tipe data *object*.

Deklarasi:

ArrayList var = new ArrayList();

Menambahkan Data Baru pada ArrayList

Memasukkan data dengan cara memanggil *method* ‘.add’. Data akan tersimpan secara sistematis di dalam sebuah ArrayList.

Memanggil Data pada ArrayList

Memanggil data dengan *method* ‘.get’ pada variabel ArrayList sesuai *index* data. Karena ArrayList menggunakan *index* maka harus memasukkan nomor *index*-nya terlebih dahulu.