

MODUL PRAKTIKUM PEMROGRAMAN BERORIENTASI OBJEK

KOLEKSI DATA (DATA COLLECTION)

Deskripsi Singkat

Praktikum pemrograman berorientasi objek adalah praktikum yang menggunakan bahasa Java sebagai bantuan dalam memahami konsep pemrograman berorientasi objek. Materi praktikum berisi teori, latihan dan soal pemrograman.

Tujuan

1. Menggunakan array satu dimensi dan array dua dimensi.
2. Menggunakan class ArrayList.
3. Menggunakan class HashMap.

Prasyarat

Siswa telah melakukan praktikum 1-8.

Materi 1 : Array

Array merupakan Objek kontainer/ penampung yang memiliki tipe data yang sama. Array juga berupa struktur data yang menyimpan elemen yang sama berbasis indeks(index). Elemen pertama sebuah array memiliki indeks 0 dan elemen terakhirnya n-1 dimana n adalah jumlah elemen di dalam array.

Array satu dimensi adalah array yang paling umum digunakan. Array dua dimensi biasanya digunakan untuk mewakili matriks.

Sintaks array satu dimensi:

```
tipe_data[] variabel_arr = { };  
tipe_data[] variabel_arr = new tipe_data[ukuran];  
tipe_data variabel_arr[] = { };  
tipe_data variabel_arr[] = new tipe_data[ukuran];
```

Contoh:

```
int[] bil_bulat = {3, 5, 7};  
String[] warna = {"merah", "hijau", "kuning"};  
int[] tts = new int[100];
```

Sintaks array dua dimensi:

```
tipe_data[][] var_arr = {{}, {}};  
tipe_data[][] var_arr = new tipe_data[ukr_baris][ukr_kolom];  
tipe_data var_arr[][] = {{}, {}};  
tipe_data var_arr[][] = new tipe_data[ukr_baris][ukr_kolom];
```

Contoh:

```
int arr[][]={{1,2,3},{3,4,5},{5,6,7}};
```

Materi 2 : ArrayList

Collection adalah suatu objek yang bisa digunakan untuk menyimpan sekumpulan objek. Objek yang ada dalam Collection disebut elemen. Collection menyimpan elemen yang bertipe Object, sehingga berbagai tipe objek bisa disimpan dalam Collection. Class-class mengenai Collection tergabung dalam Java Collection Framework. *Collections framework* merupakan bentuk algoritma yang digunakan untuk merepresentasikan dan memanipulasi collections. Semua collections frameworks mengandung hal-hal berikut:

- **Interfaces:** memungkinkan collections dimanipulasi secara independen.
- **Implementations:** merupakan implementasi dari collection interfaces. Dan merupakan struktur data yang reusable.
- **Algorithms:** merupakan method-method yang dapat digunakan untuk melakukan proses komputasi tertentu, seperti searching (pencarian) dan sorting (pengurutan), terhadap objek yang meng-implement collection interfaces. Method-method pada Java Collections Framework adalah *polymorphic*: maksudnya **nama method** yang sama dapat digunakan pada collection interface yang sesuai. Algoritma pada Java Collections Framework memiliki fungsi yang reusable.

Class-class Collection diletakkan dalam package java.util dan mempunyai dua interface utama yaitu Collection dan Map. Mulai java 1.5 (juga dikenal sebagai J2SE 5), semua class yang termasuk Java Collection Framework adalah class generics. Untuk kompatibilitas dengan versi java sebelumnya, penggunaan generics tidak diharuskan, namun sangat disarankan.

ArrayList merupakan subclass dari AbstractList dan mengimplementasi interface List. ArrayList menggunakan array dinamis untuk menyimpan elemen. ArrayList tidak memerlukan sinkronisasi.

ArrayList memiliki kemiripan dengan array namun ArrayList berupa array yang fleksibel sebab dapat menambah kapasitas sesuai keperluan. Elemen di dalam ArrayList juga diakses dengan menggunakan indeks. Namun indeks tersebut bisa berubah jika terdapat elemen yang ditambah atau dihapus. Method utama pada ArrayList adalah: add, get, remove dan size. Detil class ArrayList serta methodnya dapat dilihat pada API: <https://docs.oracle.com/javase/8/docs/api/java/util/ArrayList.html>

Contoh:

```
List<String> list = new ArrayList<String>(c);  
  
List<String> list = new ArrayList<>(c); //Mulai JDK 7 dan terbaru
```

Materi 3 : HashMap

Perbedaan mendasar map dengan collection yang lain, untuk menyimpan objek pada Map, perlu sepasang objek, yaitu key yang bersifat unik dan nilai yang disimpan. Untuk mengakses nilai tersebut maka kita perlu mengetahui key dari nilai tersebut. Map juga dikenal sebagai dictionary/kamus. Pada saat menggunakan kamus, perlu suatu kata yang digunakan untuk pencarian. Class-class yang mengimplementasikan Map adalah Hashtable, HashMap, LinkedHashMap.

HashMap merupakan subclass dari AbstractMap dan mengimplementasi interface Map. HashMap tidak memiliki urutan. HashMap memiliki value yang berbasis key dan elemen yang ada adalah unik. Method pada class HashMap diantaranya isEmpty, clear, put, entrySet. Untuk detail class HashMap dan methodnya dapat dilihat pada API: <https://docs.oracle.com/javase/8/docs/api/java/util/HashMap.html>

Contoh:

```
Map<String, String> map = new HashMap<>();
```

LATIHAN 1

Ketik dan jalankan coding di bawah:

```
public class Array1{  
    public static void main(String[] ar){  
        //deklarasi dan inisialisasi array  
        int[] bil_bulat = {3, 5, 7};  
        String[] warna = {"merah", "hijau", "kuning"};  
  
        //menampilkan array  
        for(int i=0;i<bil_bulat.length;i++){  
            System.out.print(bil_bulat[i] + " ");  
        }  
        System.out.println();  
  
        for(int i=0;i<warna.length;i++){  
            System.out.print(warna[i] + " ");  
        }  
        System.out.println();  
  
        //deklarasi dan inisialisasi array 2D  
        int arr[][]={{1,2,3},{3,4,5},{5,6,7}};  
  
        //menampilkan array 2D
```

```
        for(int i=0;i<3;i++){
            for(int j=0;j<3;j++){
                System.out.print(arr[i][j]+" ");
            }
            System.out.println();
        }
    }
}
```

Apakah hasil outputnya ketika coding di atas dijalankan?

LATIHAN 2

Ketik dan jalankan coding di bawah:

```
import java.util.Arrays;

public class ArraySatuDimensi {

    public static void main(String[] args) {

        System.out.println("Array of primitives");
        int[] ints = {9, 6, 3};
        Arrays.sort(ints);
        for (int i = 0; i < ints.length; i++) {
            System.out.println(ints[i]);
        }

        System.out.println("Array of strings");
        String[] strings = {"Merah", "Hijau", "Biru"};
        Arrays.sort(strings);
        for (String color : strings) {
            System.out.println(color);
        }

        System.out.println("Setting an initial size");
        int[] sized = new int[10];
        for (int i = 0; i < sized.length; i++) {
            sized[i] = i * 100;
        }
        for (int value : sized) {
            System.out.println(value);
        }

        System.out.println("Copying an array");
        int[] copied = new int[5];
        System.arraycopy(sized, 5, copied, 0, 5);
        for (int value : copied) {
            System.out.println(value);
        }
    }
}
```

Apakah hasil outputnya ketika coding di atas dijalankan?

LATIHAN 3

Ketik dan jalankan coding di bawah:

```
public class ArrayMultiDimensi {  
  
    public static void main(String[] args) {  
  
        String[][] provinsi = new String[3][2];  
        provinsi[0][0] = "Aceh";  
        provinsi[0][1] = "Banda Aceh";  
        provinsi[1][0] = "Maluku Utara";  
        provinsi[1][1] = "Sofifi";  
        provinsi[2][0] = "Papua Barat";  
        provinsi[2][1] = "Manokwari";  
  
        for (int i = 0; i < provinsi.length; i++) {  
            StringBuilder sb = new StringBuilder();  
            sb.append("Ibukota ")  
                .append(provinsi[i][0])  
                .append(" adalah ")  
                .append(provinsi[i][1])  
                .append(".");  
            System.out.println(sb);  
        }  
    }  
}
```

Apakah hasil outputnya ketika coding di atas dijalankan? Pahami output yang terhasil.

LATIHAN 4

Ketik dan jalankan coding di bawah:

```
import java.util.ArrayList;  
import java.util.Iterator;  
import java.util.List;  
  
class Mahasiswa {  
    String npm;  
    String nama;  
    int angkatan;  
  
    Mahasiswa (String npm, String nama, int angkatan){  
        this.npm = npm;  
        this.nama = nama;  
        this.angkatan = angkatan;  
    }  
}  
  
public class ArrayListContoh {
```

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

    //Membuat object dari kelas Mahasiswa
    Mahasiswa mhs1 = new Mahasiswa("1608107010021", "Fulan 21", 2016);
    Mahasiswa mhs2 = new Mahasiswa("1608107010002", "Fulan 2", 2016);
    Mahasiswa mhs3 = new Mahasiswa("1608107010046", "Fulan 46", 2016);

    //Membuat ArrayList
    List<Mahasiswa> mahasiswa = new ArrayList<>();
    mahasiswa.add(mhs1);
    mahasiswa.add(mhs2);
    mahasiswa.add(mhs3);

    //Perulangan menggunakan Interface Iterator
    Iterator itr = mahasiswa.iterator();

    while (itr.hasNext()){
        Mahasiswa m = (Mahasiswa) itr.next();
        System.out.println(m.npm + " " + m.nama + " " + m.angkatan);
    }
}
```

Apakah hasil outputnya ketika coding di atas dijalankan? Pahami output yang terhasil.

LATIHAN 5

Ketik dan jalankan coding di bawah:

```
import java.util.HashMap;
import java.util.Map;

class Buku {
    int id;
    String nama, pengarang, publisher;
    int jumlah;
    public Buku(int id, String nama, String pengarang, String publisher, int
jumlah) {
        this.id = id;
        this.nama = nama;
        this.pengarang = pengarang;
        this.publisher = publisher;
        this.jumlah = jumlah;
    }
}

public class Main {
    public static void main(String[] args) {
        //Membuat map Buku
        Map<Integer, Buku> map=new HashMap<Integer, Buku>();

        //Membuat objek Buku
        Buku b1=new Buku(101,"Design Patterns: Elements of Reusable Object-
Oriented Software","Erich Gamma","Addison-Wesley Professional",8);
```

```
Buku b2=new Buku(102,"Problem Frames: Analysing & Structuring  
Software Development Problems","Michael Jackson","Addison-Wesley  
Professional",4);
```

```
Buku b3=new Buku(103,"Software Architecture in Practice","Len  
Bass","Addison-Wesley Professional",6);
```

```
//Menambahkan Buku ke map  
map.put(1,b1);  
map.put(2,b2);  
map.put(3,b3);  
  
//Proses Traversing map  
for(Map.Entry<Integer, Buku> entry:map.entrySet()){  
    int key=entry.getKey();  
    Buku b=entry.getValue();  
    System.out.println(key+" Details:");  
    System.out.println(b.id+" "+b.nama+" "+b.pengarang+"  
"+b.publisher+" "+b.jumlah);  
}  
}
```

Apakah hasil outputnya ketika coding di atas dijalankan? Pahami output yang terhasil.

SOAL-SOAL

1. Gunakan contoh pada Latihan 4 lalu coba gunakan method ArrayList berikut:
 - a. addAll()
 - b. removeAll()
 - c. isEmpty()
 - d. size()
 - e. contains()
 - f. get()