

Map

Map adalah struktur data dalam bentuk pasangan key-value. Map sama dengan “associative array “ dalam bahasa PHP. Objek disimpan di Map sebagai value menggunakan key yang harus unik dan harus berupa objek juga. Salah satu implementasi dari interface Map adalah class HashMap.

Codelab Map

Untuk lebih mengenal Map mari kita langsung coding.

1. Bukalah kembali proyek Collection.



2. Buatlah kelas baru dengan nama **Planet** dan masukkan kode berikut ke dalamnya:

```
1. package com.dicoding.javafundamental.collection;
2.
3. class Planet {
4.     private String name;
5.     private double mass;
6.
7.     public Planet(String name, double mass) {
8.         this.name = name;
9.         this.mass = mass;
10.    }
11.
12.    @Override
13.    public String toString() {
14.        return "Planet " + name + ", mass: " + mass;
15.    }
```

3. Buatlah kelas **MapPlanet** dan tambahkan kode berikut di dalamnya:

```
1. package com.dicoding.javafundamental.collection;
2.
3. import java.util.HashMap;
4. import java.util.Map;
5.
6. public class MapPlanet{
7.
8.     public static void main(String[] args) {
9.         // menggunakan HashMap
10.        Map<String, Planet> planets = new HashMap();
11.        planets.put("key-1", new Planet("Mercury", 0.06)); // method put() untuk menambahkan objek ke Map
12.        planets.put("key-2", new Planet("Venus", 0.82));
13.        planets.put("key-3", new Planet("Earth", 1.00));
14.        planets.put("key-4", new Planet("Mars", 0.11));
15.        planets.put("key-4", new Planet("Mars-X", 0.11)); // menambahkan ke Map dengan key yang sama
```

4. Outputnya akan seperti di bawah ini. Perhatikan. Jika menambahkan dengan key yang sama maka value akan diganti. Lebih lanjut, sama halnya pada Set, objek yang dimasukkan ke Map tidak teratur urutannya.

```
Map planets awal: (size = 4)
key-1 : Planet Mercury, mass: 0.06
key-4 : Planet Mars-X, mass: 0.11
key-3 : Planet Earth, mass: 1.0
key-2 : Planet Venus, mass: 0.82
Map planets akhir: (size = 3)
Planet Mercury, mass: 0.06
Planet Mars-X, mass: 0.11
Planet Earth, mass: 1.0
```

Pada dasarnya Collection adalah struktur data. Makin kompleks struktur datanya maka akan dibutuhkan implementasi Collection yang lebih kompleks lagi. Pada materi ini kita hanya membahas Collection yang sudah ada di JDK tanpa menambahkan library apapun. Di luar sana ada banyak library Collection untuk keperluan yang kompleks misal Apache Common Collection dan Google Collection (sekarang menjadi bagian dari Guava).

Baca beberapa tautan di bawah ini untuk menambah pengetahuan Anda.

- <https://commons.apache.org/proper/commons-collections/>
- <https://code.google.com/archive/p/google-collections/>
- <https://github.com/google/guava>

