

## Jurnal 06: Hashing dan Map

Kerjakanlah soal berikut. Jawaban dinilai berdasarkan dua kriteria: a. Kode sesuai dengan permintaan soal, dan b. Sesi tanya-jawab dengan Asprak. Kerjakan sesuai dengan prinsip Java Coding Style.

### A. Hashing

Pada hari esok, sebuah pertandingan **3vs3** bola akan diadakan pada sore hari, tetapi tim Ego jinpachi tidak mempunyai program untuk menentukan pemain cadangannya dengan efisien. Ego jinpachi kesusahan untuk membuat sistem yang bisa memilih pemain cadangan dengan efisien. Bantulah Ego jinpachi untuk membuat program tersebut!

Setiap pemain hanya perlu memiliki Nama Pemain tersebut, dan pemain utama harus diisi terlebih dahulu, baru kalian bisa menambahkan pemain cadangannya, karena itu kita akan menggunakan konsep **hashing** dengan teknik **Separate Chaining**.

#### Persyaratan Implementasi :

- Cukup tiga kelas. Kelas HashSet, Kelas Node, dan Kelas Main.
- Menggunakan Hash Code bawaan Java Library.
- Nama Pemain adalah key.
- 3 Pemain utama **diharuskan** terisi terlebih dahulu sebelum pemain cadangan. (Maksudnya setiap indeks harus terisi terlebih dahulu sebelum ada **collision** yang terjadi).
- Panjang tabel adalah 3.
- Dapat mencetak seluruh anggota tim.
- Dapat mencetak khusus pemain utama saja.
- Dapat mencetak khusus pemain cadangan saja.

#### Challenge :

Buatlah hal yang sama, tetapi dengan pertandingan 5vs5  
Cari terlebih dahulu 5 Pemain Utama, dan selanjutnya bebas jumlah pemain cadangannya.

## **B. Map**

Sebentar lagi akan ada pendaftaran proyek “Blue Lock” untuk mencari penyerang terbaik di tingkat nasional, agar memudahkan manajemen pemain, anda sebagai seorang programmer diminta untuk membuat suatu sistem untuk menyimpan data pemain dalam konsep hashing dalam kelas bentukan dan hashmap. Nantinya hashmap akan berfungsi untuk menyimpan data pemain sebagai value dengan kode unik atau key yang dapat mereferensikan nama, umur, posisi, yang dimana key akan dibuat berdasarkan hashing dari nomor pemain.

### **Persyaratan Implementasi :**

- Minimal memiliki 2 kelas. Kelas Pemain (POJO), dan kelas Main
- Menggunakan Hash Code buatan sendiri yang dibuat dalam kelas pojo
- Key berasal dari HashCode nomor pemain
- Minimal memiliki 3 Data berbeda
- Memiliki setidaknya 1 Data yang sama
- Dapat mencetak semua isi hashmap
- Dapat mendeteksi adanya duplikasi data dan mencetak data yang terduplikasi

### **Challenge :**

Buatlah agar dapat menemukan data pemain berdasarkan key yang diinginkan

## Java Coding Style

Ikuti gaya penulisan berikut di kode program agar mendapat nilai maksimal

1. Setiap nama kelas, variabel dan konstanta harus dapat menggambarkan isinya.

Misal, membuat variabel untuk menyimpan jumlah penumpang di bus. Deklarasi:

```
int penumpang = 0;           // Bagus, OK
int p = 0;                   // Tidak OK
```

2. Penulisan nama harus menggunakan huruf besar/kecil yang sesuai.

Nama kelas: UpperCamelCase. Misal: PenumpangBus

Nama variabel: lowerCamelCase. Misal: penumpangBus

Nama konstanta: CAPITAL\_CASE. Misal: PENUMPANG\_BUS

**Nama method: lowerCamelCase. Misal: hitungPenumpangBus**

3. Penggunaan tab/spasi ketika berbeda blok program, WAJIB masuk 1 tab atau 4 spasi ke dalam.

```
public class PenumpangBus {
    public static void main(String[] args) {
        int penumpang = 0;
        penumpang = penumpang + 5;
        penumpang = penumpang - 4 + 2;
        penumpang = penumpang - 1;
        penumpang = penumpang - 2 + 3;
        penumpang = penumpang - 2 + 5;
        penumpang = penumpang - 1 + 3;
        System.out.println(penumpang);
    }
}
```

4. Gunakan komentar seperlunya.

Javadoc comment (yang warna biru) hanya untuk mengomentari kelas, method dan variable

```
/** Jumlah penumpang di bus */  
int penumpang = 0;  
  
/* Ada 5 orang penumpang naik ke bus */  
penumpang = penumpang + 5;  
  
// Penumpang turun 4 orang, naik 2  
penumpang = penumpang - 4 + 2;
```