

Jobsheet Minggu ke-12

Collection (List, Set, Map) dan Database

Nama : Reza Kurniawan

Kelas : SIB 2E

1. Tujuan Pembelajaran

- Memahami cara penyimpanan objek menggunakan Collection dan Map.
- Mengetahui pengelompokan dari Collection.
- Mengetahui perbedaan dari interface Set, List dan Map.
- Mengetahui penggunaan class-class dari interface Set, List, dan Map.
- Memahami koneksi database menggunakan JDBC dan JDBC API

2. Materi

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 object bisa disimpan dalam Collection. Class-class mengenai Collection tergabung dalam Java Collection Framework. Class-class Collection diletakkan dalam package java.util dan mempunyai dua interface utama yaitu **Collection**.

Collection terbagi menjadi 3 kelompok yaitu

a. Set

Set mengikuti model himpunan, dimana objek/anggota yang tersimpan dalam Set harus unik. Urutan maupun letak dari anggota tidak penting, hanya keberadaan anggota saja yang penting. Kelas konkret yang mengimplementasikan **Set** harus memastikan bahwa tidak terdapat elemen duplikat yang dapat ditambahkan ke dalam set. Yaitu, tidak terdapat dua elemen **e1** dan **e2** yang berada di dalam set yang membuat **e1.equals(e2)** bernilai **true**. Class-Class yang mengimplementasikan interface Set adalah HashSet.

HashSet dapat digunakan untuk menyimpan elemen-elemen bebas-duplikat. Kelas HashSet merupakan suatu kelas konkret yang mengimplementasikan Set. Pembuatan objek HashSet adalah sebagai berikut:

```
Set <nama_objek_HashSet> = new HashSet();
```

b. List

List digunakan untuk menyimpan sekumpulan objek berdasarkan urutan masuk (ordered) dan menerima duplikat. Cara penyimpanannya seperti array, oleh sebab itu memiliki posisi awal dan posisi akhir, menyisipkan objek pada posisi tertentu, mengakses dan menghapus isi list, dimana semua proses ini selalu didasarkan pada urutannya. Class-class yang

mengimplementasikan interface List adalah Vector, Stack, Linked List dan Array List. Pada jobsheet ini yang akan dibahas adalah ArrayList.

ArrayList digunakan untuk membuat array yang ukurannya dinamis. Berbeda dengan array biasa yang ukurannya harus ditentukan di awal deklarasi array, dengan ArrayList, ukurannya akan fleksibel tergantung banyaknya elemen yang dimasukkan. Pendeklarasian object ArrayList sebaiknya diikuti dengan nama class yang akan dimasukkan dalam List tersebut. Tujuannya agar method dan property dari setiap object dalam ArrayList dapat diakses secara langsung. Pembuatan objek ArrayList adalah sebagai berikut:

ArrayList <nama_objek_arraylist>=new ArrayList();

c. Map

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. Pada jobsheet ini yang akan dibahas adalah HashMap.

HashMap adalah class implementasi dari Map, Map itu sendiri adalah interface yang memiliki fungsi untuk memetakan nilai dengan key unik. HashMap berfungsi sebagai *memory record management*, dimana setiap *record* dapat disimpan dalam sebuah Map. kemudian setiap Map diletakkan pada vektor, list atau set yang masih turunan dari collection. Pembuatan objek HashMap adalah sebagai berikut:

HashMap <nama_objek_HashMap> = new HashMap();

d. JDBC API

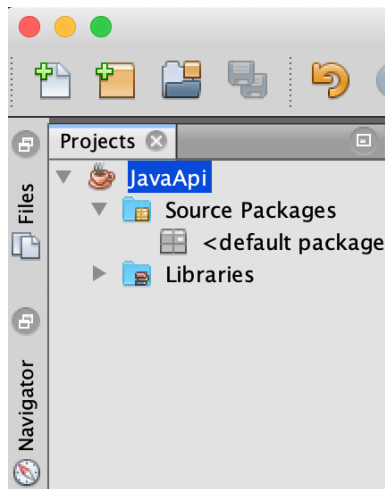
JDBC API merupakan Java Database Connectivity Application Programming Interface (JDBC API). Pada dasarnya JDBC API terdiri dari satu set kelas dan interface yang digunakan untuk berinteraksi dengan database dari aplikasi Java. Umumnya, JDBC API melakukan 3 (tiga) fungsi berikut :

- Membangun koneksi antara aplikasi Java dan database
- Membangun dan mengeksekusi query
- Memproses hasil

3. Percobaan 1 (Set)

A. Langkah Percobaan (HashSet)

1. Bukalah program Netbeans IDE yang sudah terinstal di komputer anda!
2. Buatlah project baru dengan nama JavaApi, ilustrasi project dapat dilihat pada gambar dibawah ini:



3. Setelah itu buatlah package dengan nama identifier.percobaan1, misalnya : arie.percobaan1.
4. Pada package tersebut buatlah java class dengan nama DemoHashSet.
5. Tambahkan source code yang ada di method main() kedalam class yang sudah anda buat!

```
package arie.percobaan1;

import java.util.HashSet;
import java.util.Iterator;
import java.util.Set;

public class CobaHashSet {
    public static void main(String[] args) {
        Set mSetCity = new HashSet();
        mSetCity.add("Malang");
        mSetCity.add("Banyuwangi");
        mSetCity.add("Jogjakarta");
        mSetCity.add("Batu");

        System.out.println(mSetCity);
        Iterator<String> mIterator = mSetCity.iterator();

        while (mIterator.hasNext()){
            System.out.println(mIterator.next().toLowerCase());
        }
    }
}
```

6. Jalankan program diatas, dan amati apa yang terjadi!

B. Pertanyaan

1. Apakah fungsi **import java.util.*;** pada program diatas!

Jawaban :

import java.util.HashSet; mengimpor kelas HashSet yang merupakan implementasi dari Set yang menggunakan tabel hash untuk menyimpan elemen-elemennya.

import java.util.Iterator; mengimpor antarmuka Iterator yang digunakan untuk mengulang elemen-elemen dalam sebuah koleksi.

import java.util.Set; mengimpor antarmuka Set yang merupakan koleksi yang tidak mengizinkan elemen duplikat.

2. Pada baris program keberapakah yang berfungsi untuk menciptakan object HashSet?

Jawaban :

```
Set mSetCity = new HashSet();
```

3. Apakah fungsi potongan program dibawah ini pada percobaan 1!

```
mSetCity.add("Malang");  
mSetCity.add("Banyuwangi");  
mSetCity.add("Jogjakarta");  
mSetCity.add("Batu");
```

Jawaban : digunakan untuk menyimpan nama-nama kota.

4. Tambahkan `set.add("Malang")`; kemudian jalankan program! Amati hasilnya dan jelaskan apa yang terjadi!

Jawaban :

```
[Banyuwangi, Malang, Jogjakarta, Batu]  
banyuwangi  
malang  
jogjakarta  
batu
```

Jawaban : karena Salah satu sifat utama dari HashSet adalah bahwa ia tidak mengizinkan duplikasi elemen.

5. Jelaskan fungsi potongan program dibawah ini pada percobaan 1!

```
Iterator<String> mIterator = mSetCity.iterator();  
  
while (mIterator.hasNext()){  
    System.out.println(mIterator.next().toLowerCase());  
}
```

Jawaban : untuk menelusuri elemen-elemen dalam HashSet dan mencetak setiap elemen dalam huruf kecil

4. Percobaan 2 (List)

A. Langkah Percobaan (ArrayList)

1. Buatlah package dengan nama identifier.percobaan2 pada project yang sudah anda buat sebelumnya, misalnya : arie.percobaan2.
2. Tambahkan source code berikut pada class yang sudah anda buat!

```
package arie.percobaan2;  
  
import java.util.ArrayList;  
import java.util.List;  
  
public class CobaArrayList {  
    public static void main(String[] args) {  
        List mListCountry = new ArrayList();  
        mListCountry.add("Indonesia");  
        mListCountry.add("Malaysia");  
        mListCountry.add("German");  
        System.out.println(mListCountry);  
        System.out.format("List index 0 = %s\n", mListCountry.get(0));  
        System.out.format("List index 2 = %s\n", mListCountry.get(2));  
    }  
}
```

3. Jalankan program diatas!

B. Pertanyaan

1. Apakah fungsi potongan program dibawah ini!

```
System.out.println(mListCountry);  
System.out.format("List index 0 = %s\n", mListCountry.get(0));  
System.out.format("List index 2 = %s\n", mListCountry.get(2));
```

Jawaban :

System.out.println(mListCountry); “untuk menampilkan seluruh isi array yaitu Indonesia, malaysia, dan german”

System.out.format("List index 0 = %s\n", mListCountry.get(0)); “untuk menampilkan isi array yang berada di index 0 yaitu indonesia

System.out.format("List index 2 = %s\n", mListCountry.get(2)); “untuk menampilkan isi array yang berada di index 2 yaitu german

2. Ganti potongan program pada soal no 1 menjadi sebagai berikut

```
System.out.println("Country List");  
Iterator mIterator = mListCountry.iterator();  
while(mIterator.hasNext()){  
    Object element = mIterator.next();  
    System.out.format("%s ", element);  
}
```

Kemudian jalankan program tersebut!

Jawaban :

```
Country List  
IndonesiaMalaysiaGerman
```

3. Jelaskan perbedaan menampilkan data pada ArrayList menggunakan potongan program pada soal no 1 dan no 2!

Jawaban :

Pada soal no 1 mencetak elemen-elemen dalam bentuk list yang dipisahkan oleh koma dan diapit oleh tanda kurung siku (misalnya, [element1, element2, element3]).

Jika pada soal no 2 Tidak ada karakter pemisah atau newline yang ditambahkan di antara elemen-elemen yang ditampilkan, sehingga semua elemen akan ditampilkan berdekatan.

5. Percobaan 3 (Map)

A. Langkah Percobaan

1. Buatlah package dengan nama identifier.percobaan3 pada project yang sudah anda buat sebelumnya, misalnya : arie.percobaan3.
2. Buatlah class baru dengan nama DemoHashMap. Tambahkan source code dibawah ini kedalam class!

```

package arie.percobaan3;

import java.util.HashMap;

public class DemoHashMap {
    public static void main(String[] args) {
        HashMap hMapItem = new HashMap();
        hMapItem.put("1", "Biskuit");
        hMapItem.put("2", "Shampoo");
        hMapItem.put("3", "Soap");
        System.out.println(hMapItem);
        System.out.format("HashMap Item Tolal : %d\n", hMapItem.size());

        Object mObject = hMapItem.remove("1");
        System.out.format("%s remove from Hashmap\n", mObject);
        System.out.format("HashMap Item Tolal : %d\n", hMapItem.size());
        System.out.println(hMapItem);

        hMapItem.clear();
        System.out.format("HashMap Item Tolal : %d\n", hMapItem.size());
    }
}

```

3. Jalankan program diatas, maka outputnya adalah sebagai berikut:

```

run:
{1=Biskuit, 2=Shampoo, 3=Soap}
HashMap Item Tolal : 3

Biskuit remove from Hashmap
HashMap Item Tolal : 2
{2=Shampoo, 3=Soap}
HashMap Item Tolal : 0
BUILD SUCCESSFUL (total time: 0 seconds)

```

B. Pertanyaan

1. Jelaskan fungsi **hMapItem.put("1", "Biskuit")** pada program!
Jawaban : untuk menambahkan pasangan kunci ("1") dan nilai ("Biskuit") ke dalam HashMap
2. Jelaskan fungsi **hMapItem.size()** pada program!
Jawaban : memberikan informasi mengenai berapa banyak item yang saat ini tersimpan dalam HashMap.
3. Jelaskan fungsi **hMapItem.remove("1")** pada program!
Jawaban : digunakan untuk menghapus pasangan kunci-nilai yang memiliki kunci "1" dari HashMap
4. Jelaskan fungsi **hMapItem.clear()** pada program!
Jawaban : Setelah pemanggilan fungsi ini, HashMap akan menjadi kosong dan tidak akan mengandung item apapun. Ukuran HashMap (hMapItem.size()) akan menjadi nol.
5. Tambahkan kode program yang di blok pada program yang sudah anda buat!

```

public class DemoHashMap {
    public static void main(String[] args) {
        HashMap hMapItem = new HashMap();
        hMapItem.put("1", "Biskuit");
        hMapItem.put("2", "Shampoo");
        hMapItem.put("3", "Soap");
        System.out.println(hMapItem);

        Collection mCollection = hMapItem.values();
        Iterator mIterator = mCollection.iterator();
        while (mIterator.hasNext()) {
            System.out.println(mIterator.next());
        }

        System.out.format("HashMap Item Total : %d\n\n", hMapItem.size());

        Object mObject = hMapItem.remove("1");
        System.out.format("%s remove from Hashmap\n", mObject);
        System.out.format("HashMap Item Total : %d\n", hMapItem.size());
        System.out.println(hMapItem);

        hMapItem.clear();
        System.out.format("HashMap Item Total : %d\n", hMapItem.size());
    }
}

```

6. Jalankan program dan amati apa yang terjadi!

Jawaban :

```

{1=Biskuit, 2=Shampoo, 3=Soap}
Biskuit
Shampoo
Soap
HashMap Item Total : 3

Biskuit remove from Hashmap
HashMap Item Total: 2

{2=Shampoo, 3=Soap}
HashMap Item Total: 0

```

7. Apakah perbedaan program sebelumnya dan setelah ditambahkan kode program pada soal no 5 diatas? Jelaskan!

Jawaban : Versi sebelum tidak mencetak setiap item satu per satu sebelum menghapus item.
Versi sesudah mencetak setiap item satu per satu