

***LAPORAN TUGAS PRAKTIKUM PEMROGRAMAN
BERORIENTASI OBJEK***

Pertemuan Ke-: 12	
Pembahasan: Java API dan Database	
NIM: 1841720024	Dosen Pengampu: Septian Enggar Sukmana SPd., MT.
Nama Mahasiswa: Alwy Abdullah	Nilai:

TUJUAN PRAKTIKUM (10 points)

Petunjuk: tulis tujuan dari praktikum yang telah dilaksanakan

Memahami cara penyimpanan objek menggunakan Collection dan Map.

1. Mengetahui pengelompokan dari Collection.
2. Mengetahui perbedaan dari interface Set, List dan Map.
3. Mengetahui penggunaan class-class dari interface Set, List, dan Map.
4. Memahami koneksi database menggunakan JDBC dan JDBC API

JAWABAN PERTANYAAN (30 points)

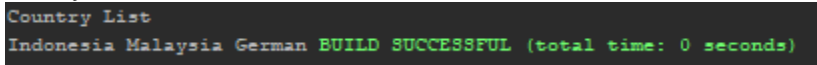
Petunjuk: jawaban dari pertanyaan yang terdapat pada modul 6 dituliskan pada bagian kolom kotak yang telah disediakan

PERCOBAAN 1

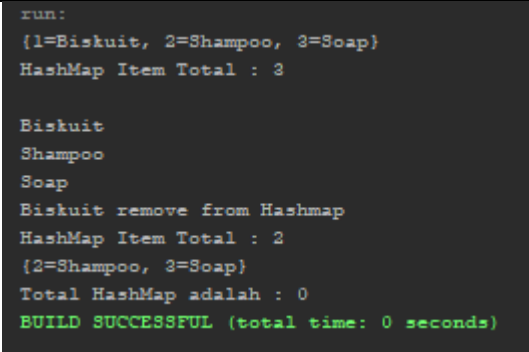
1	Untuk mengimport semua library yang ada pada program (NetBeans)
2	<pre>Set mSetCity = new HashSet();</pre> <p>Pada baris ini kita menciptakan object HashSet</p>
3	<p>Fungsi potongan kode ini :</p> <pre>mSetCity.add("Malang"); mSetCity.add("Banyuwangi"); mSetCity.add("Jogjakarta"); mSetCity.add("Batu");</pre> <p>Digunakan untuk menambahkan data dalam object mSetCity</p>
4	<pre>mSetCity.add("Malang"); mSetCity.add("Banyuwangi"); mSetCity.add("Jogjakarta"); mSetCity.add("Batu"); mSetCity.add("Malang"); ----- [Banyuwangi, Malang, Jogjakarta, Batu] banyuwangi malang jogjakarta batu BUILD SUCCESSFUL (total time: 0 seconds)</pre> <p>Program tidak error akan tetapi data yang ditambahkan terakhir (malang) hanya akan ditampilkan sekali karena Set /HashSet hanya menerima data yang sama 1 saja</p>
5	<pre>Iterator<String> iterator = mSetCity.iterator(); while (iterator.hasNext()) { System.out.print(iterator.next().toLowerCase() + " "); }</pre> <p>Fungsi potongan kode diatas adalah untuk menambahkan Interface Iterator dengan tipe data String yang setelah itu digunakan untuk melakukan perulangan dengan kondisi apakah iterator.hasNext() maksud dari kode ini apakah untuk melakukan pengecekan apakah HashSet masih memiliki data berikutnya jika ada maka akan di print</p>

PERCOBAAN 2

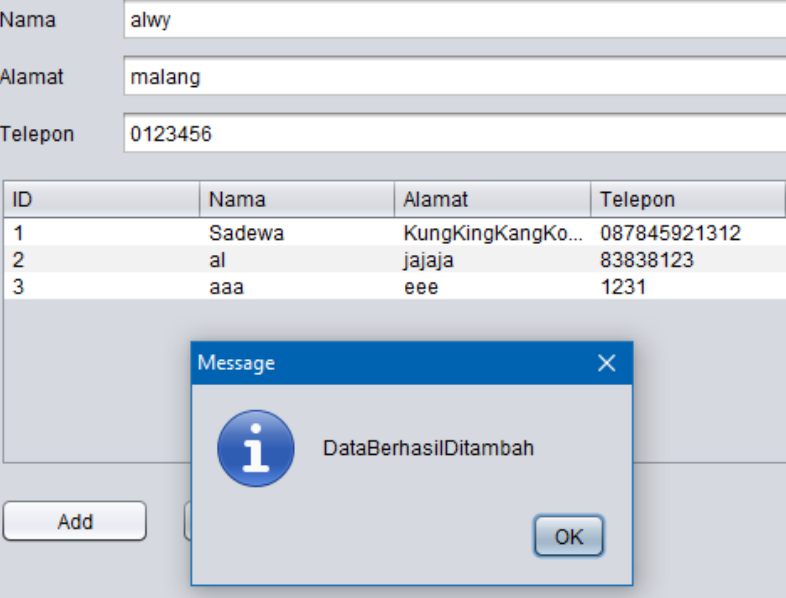
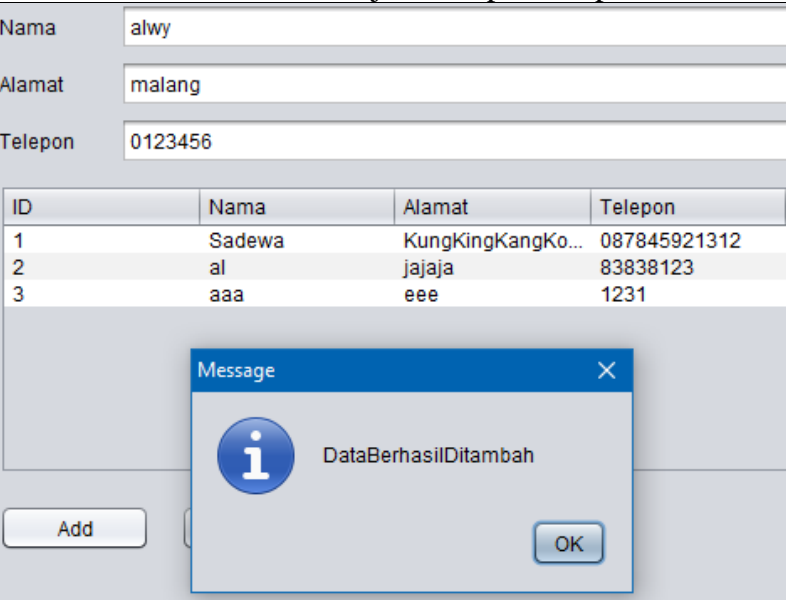
1	<pre>System.out.println(mListCountry); System.out.format("List index 0 = %s\n", mListCountry.get(0)); System.out.format("List index 2 = %s\n", mListCountry.get(2));</pre>
---	--


	<p>Pertama kita menampilkan seluruh data dari ArrayList yang telah kita masukkan setelah itu kita menampilkan data pertama dari index ke 0 dengan memanggil mListCountry.get(0)</p> <p>Setelah itu kita menampilkan data ketiga dari index ke 2 dengan memanggil mListCountry.get(2)</p>
2	<pre>System.out.println("Country List"); Iterator mIterator = mListCountry.iterator(); while (mIterator.hasNext()) { Object element = mIterator.next(); System.out.format("%s ",element); } </pre> 
3	<p>Perbedaan dari keduanya adalah cara memanggil atau cara menampilkan data tersebut pada soal no 1 kita dapat memanggilnya secara langsung</p> <p>Sedangkan pada soal no 2 kita harus menggunakan Object mIterator dari Interface Iterator setelah itu kita lakukan perulangan untuk menampilkan data yang telah kita masukkan tersebut</p>

PERCOBAAN 3

1	Digunakan untuk memasukkan data dengan key = 1 dan value = Biskuit
2	Digunakan untuk menampilkan jumlah data yang ada pada HashMap
3	Digunakan untuk menghapus data dengan key 1 pada HashMap
4	Digunakan untuk menghilangkan seluruh data dari HashMap
5	<pre>Collection mCollection = hMapItem.values(); Iterator mIterator = mCollection.iterator(); while (mIterator.hasNext()) { System.out.println(mIterator.next()); } </pre>
6	
7	<p>Setelah ditambahkan kode program tersebut maka akan ditampilkan seluruh data yang ada pada HashMap sesuai dengan urutan memasukkan data pada HashMap dan yang ditampilkan hanya Value tidak dengan key</p>

PERCOBAAN 5

5	
6	<p>Pertama membuka method <code>buka_koneksi()</code> yang digunakan untuk menghubungkan database ke mysql, kemudian menginsertkan nama, alamat, telp yang sudah terhubung dengan tabel anggota pada database mysql. Kemudian melakukan trycatch yang isinya berupa instansiasi <code>PreparedStatement</code> kemudian melakukan <code>executeUpdate</code> lalu <code>close</code>. jika data berhasil ditambahkan maka akan muncul pesan “Data Berhasil Ditambah” melalui <code>JOptionPane.showMessageDialog</code> jika gagal maka akan muncul pesan “Terjadi Kesalahan”.</p>
9	<p>Pertama membuka method <code>buka_koneksi()</code> yang digunakan untuk menghubungkan database ke mysql, kemudian <code>executeQuery(sql)</code> untuk melakukan query <code>select * from anggota</code>. Dan melakukan perulangan untuk menaruh data ke dalam object satu per satu pada tabel</p>
10	 <p>Setelah di refresh :</p>



—

□

×

Nama

alwy

Alamat

malang

Telepon

0123456

ID	Nama	Alamat	Telepon
1	Sadewa	KungKingKangKo...	087845921312
2	al	jajaja	83838123
3	aaa	eee	1231
4	alwy	malang	0123456

Add

Refresh

KODE PROGRAM DAN PENJELASAN TIAP METHODNYA (30 points)

TUGAS

Nama Class:	
Penjelasan:	
Method	Penjelasan

Petunjuk: tempatkan kode program di dalam kotak yang telah disediakan dan berikan penjelasan tentang kegunaan dari method-method yang telah dibuat, jika class lebih dari satu, buat dalam kotak yang berbeda (kotak dapat digandakan).

HASIL (15 points)

Percobaan 1 :

```
[Banyuwangi, Malang, Jogjakarta, Batu]
banyuwangi malang jogjakarta batu BUILD SUCCESSFUL (total time: 1 second)
```

Percobaan 2 :

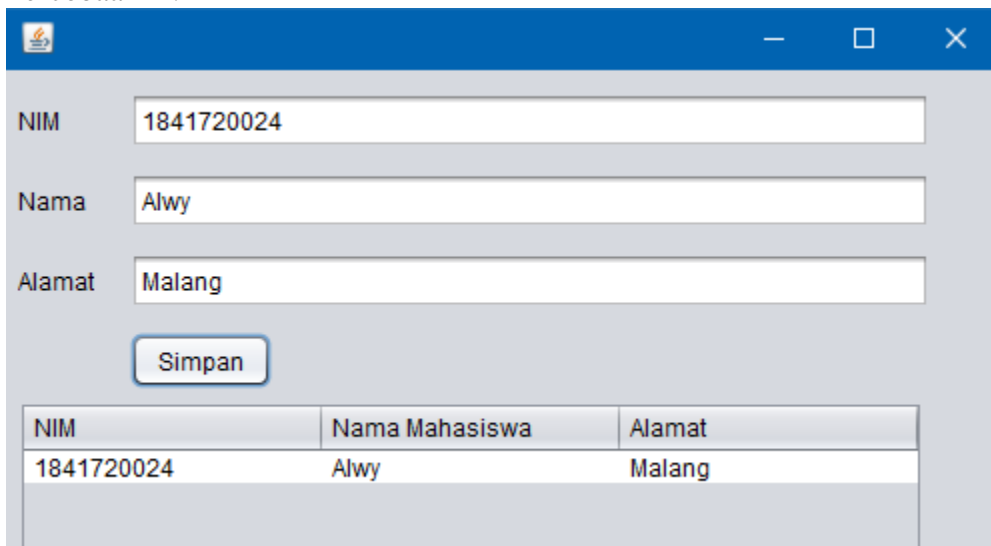
```
run:
Country List
Indonesia Malaysia German [Indonesia, Malaysia, German]
List index 0 = Indonesia
List index 2 = German
BUILD SUCCESSFUL (total time: 0 seconds)
```

Percobaan 3 :

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

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

Percobaan 4 :



The screenshot shows a Java Swing application window with a blue title bar. Inside the window, there are three text input fields labeled 'NIM', 'Nama', and 'Alamat'. The 'NIM' field contains '1841720024', the 'Nama' field contains 'Alwy', and the 'Alamat' field contains 'Malang'. Below these fields is a button labeled 'Simpan'. At the bottom of the window, there is a table with three columns: 'NIM', 'Nama Mahasiswa', and 'Alamat'. The table contains one row of data: '1841720024', 'Alwy', and 'Malang'.

NIM	Nama Mahasiswa	Alamat
1841720024	Alwy	Malang

Percobaan 5 :


Nama

Alamat

Telepon

ID	Nama	Alamat	Telepon
1	Sadewa	KungKingKangKo...	087845921312
2	al	jajaja	83838123
3	aaa	eee	1231

Message

 DataBerhasilDitambah

OK

Add

Nama

Alamat

Telepon

ID	Nama	Alamat	Telepon
1	Sadewa	KungKingKangKo...	087845921312
2	al	jajaja	83838123
3	aaa	eee	1231
4	alwy	malang	0123456

Add Refresh

KESIMPULAN (15 points)

- Collection adalah suatu obyek yang bisa Collection adalah suatu obyek yang bisa digunakan untuk menyimpan sekumpulan obyek. Obyek yang ada dalam Collection disebut elemen. Collection menyimpan elemen yang bertipe Object, sehingga berbagai tipe obyek bisa disimpan dalam Collection.

Collection terbagi menjadi 3 kelompok yaitu :

- List: sekumpulan obyek yang berdasarkan urutan (ordered) dan menerima duplikat (ordered) dan menerima duplikat. Class-class yang mengimplementasikan interface List adalah Vector, Stack, Linked List dan Array List. Pembuatan objek ArrayList adalah sebagai berikut:

```
ArrayList <nama_objek_arraylist>=new ArrayList();
```

Contoh List:

- LinkedList : elemen dalam LinkedList masuk dari awal dan dihapus dari akhir.
- Vector : a growable array of object.
- ArrayList ArrayList: mirip vector, bersifat bersifat unsynchronized unsynchronized (jika multiple multiple threads mengakses object ArrayList, object ini harus synchronized secara eksternal)
- Set: sekumpulan obyek yang tidak berdasarkan urutan (unordered) dan menolak duplikat. Class-Class yang mengimplementasikan interface Set adalah HashSet.

Pembuatan objek HashSet adalah sebagai berikut:

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

- Map: object yang memetakan key dengan value. Disebut juga dengan associative array atau dictionary.

Method untuk menambah dan menghapus

- put(Object key, Object value)
- remove (Object key)

Method untuk mengambil object.

- get (Object key)

Method untuk mengambil key, value dan pasangan (key, value)

- keySet() // returns a Set

- values() // returns a Collection
- entrySet() // returns a set

Class-class yang mengimplementasikan Map adalah Hashtable, HashMap, LinkedHashMap. Pembuatan objek HashMap adalah sebagai berikut:

```
HashMap <nama_objek_HashMap> = new HashMap();
```

- 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

Tips: tuliskan kesimpulan dari hasil tugas praktikum yang telah dilakukan.