**LAPORAN PRAKTIKUM STRUKTUR DATA**

Modul ke : 1

Judul Praktikum : Array Java Collection

Hari dan Tanggal Pelaksanaan : Selasa, 19-04-2022

Tempat Pelaksanaan : Lab Desain

Dosen Pengampu Praktikum : Khoirul Umam, S.Pd, M.Kom

Nama Mahasiswa Pelaksana : Andri WIjaksono

NIM Pelaksana : 362155401206

1. **Tugas Pendahuluan**

Tuliskan hasil pengerjaan Tugas Pendahuluan pada bagian ini.

Array adalah suatu tipe data yang menyimpan beberapa nilai dalam satu variabel.

Java collection adalah suatu objek yang dapat menyimpan sebuah objek

Pada collectin java ada beberapa kelompok yaitu **List, Set,** dan **Map**

**Lits** untuk menyimpan sekumpulan obyek berdasarkan urutan masuk (ordered) dan menerima duplikat. Class untuk pengimplementasiannya yaitu **Vector**, **Stack**, **LinkedList**, dan **ArrayList.**

**Set** mengikuti model himpunan, dimana obyek/anggota yang tersimpan dalam Set harus unik.

Class-class yang mengimplementasikan interface Set adalah **HashSet**. **Interface SortedSet** merupakan subinterface dari interface Set.

Untuk mengurutkan Set dapat dilakukan dengan menggunakan class yaitu class **TreeSet**.

**Map** Perbedaaan mendasar antara Map dengan collection yang lain ialah untuk menyimpan obyek pada Map dibutuhkan sepasang obyek, yaitu key yang bersifat unik dan nilai (value) yang disimpan. Class-class yang mengimplementasikan Map adalah **Hashtable**, **HashMap**, **LinkedHashMap**, dan **TreeMap**. Untuk mengurutkan Map dapat dilakukan dengan menggunakan interface SortedMap.

1. **Kegiatan Praktikum**

Cantumkan apa saja yang dilakukan pada latihan-latihan praktikum, *source code* yang dipakai, *screen shot* hasil eksekusi kode, dan jawaban dari pertanyaan-pertanyaan yang muncul pada ti­ap kegiatan latihan.

**Latihan 1: Mendeklarasikan array, menyimpan data di dalam array, dan mengakses data dari array.**

class ArrayDemo {

    public static void main(String[] args) {

    int[] sebuahArray; // deklarasi array integer

    sebuahArray = new int[10]; // alokasi memor untuk 10 buah integer

    sebuahArray[0] = 100; // inisialiasi elemen pertama

    sebuahArray[1] = 200; // inisialiasi elemen kedua

    sebuahArray[2] = 300; // dst

    sebuahArray[3] = 400;

    sebuahArray[4] = 500;

    sebuahArray[5] = 600;

    sebuahArray[6] = 700;

    sebuahArray[7] = 800;

    sebuahArray[8] = 900;

    sebuahArray[9] = 1000;

    System.out.println("Element at index 0: " + sebuahArray[0]);

    System.out.println("Element at index 1: " + sebuahArray[1]);

    System.out.println("Element at index 2: " + sebuahArray[2]);

    System.out.println("Element at index 3: " + sebuahArray[3]);

    System.out.println("Element at index 4: " + sebuahArray[4]);

    System.out.println("Element at index 5: " + sebuahArray[5]);

    System.out.println("Element at index 6: " + sebuahArray[6]);

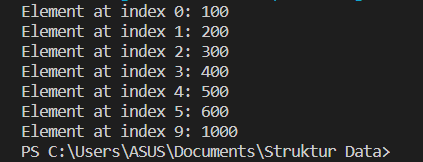
    System.out.println("Element at index 7: " + sebuahArray[7]);

    System.out.println("Element at index 8: " + sebuahArray[8]);

    System.out.println("Element at index 9: " + sebuahArray[9]);

    }

    }



4. Apa jadinya jika pada kode program di atas ditambahkan sebuah bilangan integer ke elemen array sebuahArray[10]?

Jawab:

**Latihan 2: Mengurutkan data dan menyisipkan data pada array**

import java.util.Arrays;

public class MainClass {

public static void main(String[] args) throws Exception {

int array[] = { 2, 5, -2, 6, -3, 8, 0, -7, -9, 4 };

Arrays.sort(array);

printArray("Sorted array", array);

int searchedData = 1;

int index = Arrays.binarySearch(array, searchedData);

System.out.println("Didn't find " + searchedData + " at index " +

index);

int newIndex = -index - 1;

array = insertElement(array, searchedData, newIndex);

printArray("With " + searchedData + " added", array);

}

private static void printArray(String message, int array[]) {

System.out.println(message + ": [length: " + array.length + "]");

for (int i = 0; i < array.length; i++) {

if (i != 0) {

System.out.print(", ");

}

System.out.print(array[i]);

}

System.out.println();

}

private static int[] insertElement(int original[], int element, int index)

{

int length = original.length;

int destination[] = new int[length + 1];

System.arraycopy(original, 0, destination, 0, index);

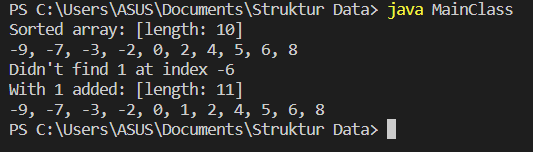
destination[index] = element;

System.arraycopy(original, index, destination, index + 1, length -

return destination;

}

}



4. Apa kegunaan dari **Arrays.sort()** dan **System.arraycopy()?**

Jawab:

**Latihan 3: Memeriksa kesamaan dua buah array**

import java.util.Arrays;

public class CekArray {

public static void main(String[] args) throws Exception {

int[] ary = { 1, 2, 3, 4, 5, 6 };

int[] ary1 = { 1, 2, 3, 4, 5, 6 };

int[] ary2 = { 1, 2, 3, 4 };

System.out.println("Apakah array 1 sama dengan array 2? " +

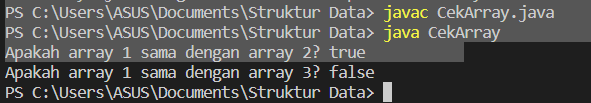
Arrays.equals(ary, ary1));

System.out.println("Apakah array 1 sama dengan array 3? " +

Arrays.equals(ary, ary2));

}

}

****

3. Apa kegunaan dari Arrays.equals() serta apa tipe data dari return value-nya?

Jawab:

**Latihan 4: Mendapatkan ukuran baris dan kolom pada array dua dimensi**

public class ArrayLength {

    public static void main(String args[]) {

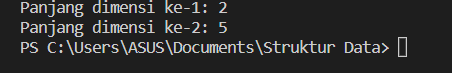
    String[][] data = new String[2][5];

    System.out.println("Panjang dimensi ke-1: " + data.length);

    System.out.println("Panjang dimensi ke-2: " + data[0].length);

    }

    }

****

3. Bagaimana cara membedakan antara dimensi pertama suatu array dengan dimensi keduanya?

Jawab:

**Latihan 5: Array dua dimensi**

public class TwoDimensions {

    public static void main(String[] args) {

    int[][] a2 = new int[10][5];

    for (int i = 0; i < a2.length; i++) {

    for (int j = 0; j < a2[i].length; j++) {

    a2[i][j] = i;

    System.out.print(" " + a2[i][j]);

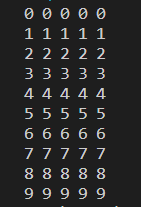
}

System.out.println("");

}

}

}

****

**Latihan 6: Non-rectangular array**

public class NonRectangularArray {

    public static void main(String[] args) {

    int myArray[][] = { { 1, 2 }, { 3 }, { 4, 5, 6 } };

    for (int i = 0; i < myArray.length; i++) {

    System.out.println("Row number " + i + " has " + myArray[i].length

    + " columns.");

    }

    for (int i = 0; i < myArray.length; i++) {

    for (int j = 0; j < myArray[i].length; j++) {

    System.out.print(myArray[i][j] + " ");

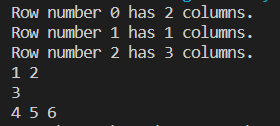
    }

    System.out.println();

    }

    }

    }

****

4. Apa yang membedakan non-rectangular array dengan array dua dimensi?

Jawab:

**Latihan 7: Penggunaan Set**

import java.util.\*;

public class SetExample {

public static void main(String[] args) {

Set set = new HashSet();

set.add("Bernadine");

set.add("Elizabeth");

set.add("Gene");

set.add("Elizabeth");

set.add("Clara");

System.out.println("Elemen pada HashSet : ");

System.out.println(set);

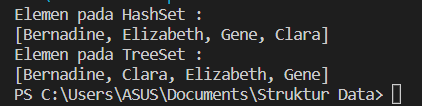
Set sortedSet = new TreeSet(set);

System.out.println("Elemen pada TreeSet : ");

System.out.println(sortedSet);

}

}

****

4. Apa perbedaan hasil antara HashSet dengan TreeSet?

Jawab: Pada Hashet kata yang sama tidak akan ditampilkan lagi, dan pada TreeSet itu mengurutkan sebuah variabel awal. TreeSet dan HashSet adalah dua kelas dalam hirarki Koleksi dan keduanya mengimplementasikan antarmuka Set. TreeSet adalah kelas yang mengimplementasikan antarmuka Set dan digunakan untuk menyimpan elemen unik dalam urutan menaik. HashSet adalah kelas yang mengimplementasikan antarmuka Set dan digunakan untuk menyimpan elemen unik menggunakan mekanisme Hashing. Jadi kesimpulannya adalah perbedaan Treeset dan Hashset adalah menyimpan elemen dalam urutan menaik sedangkan Hashset tidak menyompan elemen dalam urutan menaik.

**Latihan 8: Deteksi data duplikat menggunakan HashSet**

import java.util.\*;

public class FindDups {

public static void main(String[] args) {

Set<String> s = new HashSet<String>();

for (String a : args)

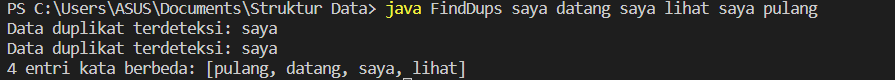
if (!s.add(a))

System.out.println("Data duplikat terdeteksi: " + a);

System.out.println(s.size() + " entri kata berbeda: " + s);

}

}

****

3. Bagaimana hasilnya? Apa yang terjadi pada data duplikat?

Jawab: iya, untuk hasil terjadi duplikat pada kata “saya”

**Contoh lainnya adalah sebagai berikut. Tuliskan kode berikut ke dalam file baru dengan nama FindDups2.java:**

import java.util.\*;

public class FindDups2 {

public static void main(String[] args) {

Set<String> uniques = new HashSet<String>();

Set<String> dups = new HashSet<String>();

for (String a : args)

if (!uniques.add(a))

dups.add(a);

// Hapus data-data pada set unique yang ditemukan

// juga pada set dups

uniques.removeAll(dups);

System.out.println("Kata-kata unik: " + uniques);

System.out.println("Kata-kata duplikat: " + dups);

}

}

****

**Latihan 9: Operasi gabungan (union) dan irisan (intersection) himpunan**

import java.util.\*;

public class SetOperation {

public static void main(String[] args) {

Set<String> s1 = new HashSet<String>();

s1.add("Australia");

s1.add("Sweden");

s1.add("Germany");

Set<String> s2 = new HashSet<String>();

s2.add("Sweden");

s2.add("France");

Set<String> union = new TreeSet<String>(s1);

union.addAll(s2); // gabungan dari s1 dan s2

printSet("Gabungan", union);

Set<String> intersect = new TreeSet<String>(s1);

intersect.retainAll(s2); // irisan dari s1 dan s2

printSet("Irisan", intersect);

}

protected static void printSet(String label, Collection<String> c) {

System.out.println("---------------- " + label + " --------------");

Iterator<String> it = c.iterator();

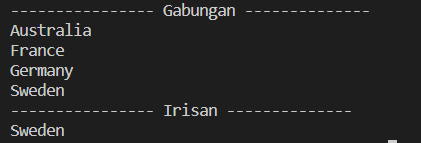
while (it.hasNext()) {

System.out.println(it.next());

}

}

}

****

3. Bagaimana hasilnya? Apa kegunaan method addAll dan retainAll pada contoh di atas?

Jawab:

**Latihan 10: Penggunaan List**

import java.util.\*;

public class ListExample {

public static void main(String[] args) {

List list = new ArrayList();

list.add("Bernadine");

list.add("Elizabeth");

list.add("Gene");

list.add("Elizabeth");

list.add("Clara");

System.out.println(list);

System.out.println("2 : " + list.get(2));

System.out.println("0 : " + list.get(0));

LinkedList queue = new LinkedList();

queue.addFirst("Bernadine");

queue.addFirst("Elizabeth");

queue.addFirst("Gene");

queue.addFirst("Elizabeth");

queue.addFirst("Clara");

System.out.println(queue);

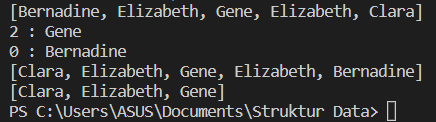
queue.removeLast();

queue.removeLast();

System.out.println(queue);

}

}

****

3. Apa yang terjadi pada data string “Elizabeth”?

Jawab:

4. Apa perbedaan antara method add dengan method addFirst?

Jawab:

**Latihan 11: Penggunaan class Vector**

import java.util.Vector;

public class VectorExample {

public static void main(String[] args) {

Vector<String> vc = new Vector<String>();

// add vector elements

vc.add("Vector Object 1");

vc.add("Vector Object 2");

vc.add("Vector Object 3");

vc.add("Vector Object 4");

vc.add("Vector Object 5");

// vc.size() inform number of elements in Vector

System.out.println("Vector Size :" + vc.size());

// get elements of Vector

for (int i = 0; i < vc.size(); i++) {

    System.out.println("Vector Element " + i + " :" + vc.get(i));

    }

    // add vector element at index

    vc.add(3, "Element at fix position");

    // vc.size() inform number of elements in Vector

    System.out.println("Vector Size :" + vc.size());

    // get elements of Vector

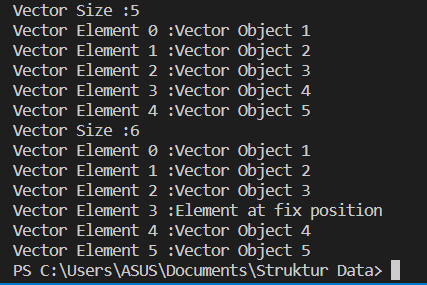
    for (int i = 0; i < vc.size(); i++) {

    System.out.println("Vector Element " + i + " :" + vc.get(i));

    }

    }

    }

****

3. Apa yang terjadi pada data-data di elemen 3 dan 4 ketika terjadi penambahan/penyisipan data baru pada elemen 3?

Jawab:

**Latihan 12: Penggunaan Iterator**

import java.util.\*;

class IteratorDemo {

public static void main(String args[]) {

// create an array list

ArrayList al = new ArrayList();

// add elements to the array list

al.add("C");

al.add("A");

al.add("E");

al.add("B");

al.add("D");

al.add("F");

// use iterator to display contents of al

System.out.print("Original contents of al: ");

Iterator itr = al.iterator();

while (itr.hasNext()) {

Object element = itr.next();

System.out.print(element + " ");}

System.out.println();

// modify objects being iterated

ListIterator litr = al.listIterator();

while (litr.hasNext()) {

Object element = litr.next();

litr.set(element + "+");

}

System.out.print("Modified contents of al: ");

itr = al.iterator();

while (itr.hasNext()) {

Object element = itr.next();

System.out.print(element + " ");

}

System.out.println();

// now, display the list backwards

System.out.print("Modified list backwards: ");

while (litr.hasPrevious()) {

Object element = litr.previous();

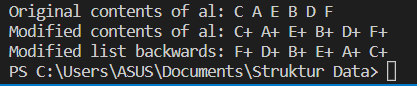
System.out.print(element + " ");

}

System.out.println();

}

}

**\**

3. Jelaskan kegunaan method hasNext, next, hasPrevious, dan previous!

Jawab:

4. Apa perbedaan antara Iterator dengan ListIterator?

Jawab:

**Latihan 13: Penggunaan Enumeration**

import java.util.Vector;

import java.util.Enumeration;

public class EnumerationTester {

public static void main(String args[]) {

Vector dayNames = new Vector();

dayNames.add("Sunday");

dayNames.add("Monday");

dayNames.add("Tuesday");

dayNames.add("Wednesday");

dayNames.add("Thursday");

dayNames.add("Friday");

dayNames.add("Saturday");

Enumeration days = dayNames.elements();

while (days.hasMoreElements())

System.out.println(days.nextElement());

}

}

****

3. Jelaskan kegunaan Enumeration!

Jawab:

**Latihan 14: Penggunaan HashMap**

import java.util.HashMap;

public class HashMapExample {

public static void main(String[] args) {

// buat objek HashMap

HashMap<String, String> ibukota = new HashMap<String, String>();

// tambahkan pasangan key-value ke dalam objek HashMap

ibukota.put("Indonesia", "Jakarta");

ibukota.put("Malaysia", "Kuala Lumpur");

ibukota.put("Inggris", "London");

cekHashMap(ibukota);

// hapus salah satu elemen HashMap

String dihapus = ibukota.remove("Malaysia");

System.out.println(dihapus + " dihapus dari HashMap");

cekHashMap(ibukota);

// bersihkan HashMap

ibukota.clear();

cekHashMap(ibukota);

}

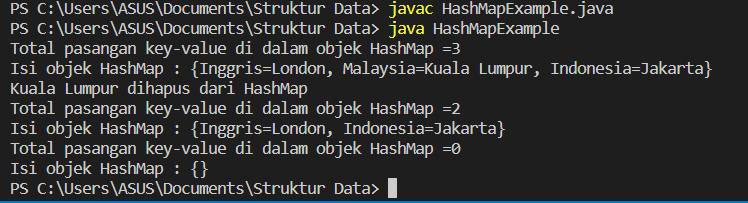
public static void cekHashMap(HashMap<String, String> obj) {

System.out.println("Total pasangan key-value di dalam objek HashMap =" + obj.size());

System.out.println("Isi objek HashMap : " + obj);

}

}

****

3. Jelaskan kegunaan method put dan remove pada HashMap!

Jawab:

4. Jelaskan kegunaan method size pada HashMap!

Jawab:

**Latihan 15: Memeriksa keberadaan suatu key dan value di dalam objek HashMap**

import java.util.HashMap;

public class KeyValueChecker {

public static void main(String[] args) {

HashMap<String, String> ibukota = new HashMap<String, String>();

ibukota.put("Indonesia", "Jakarta");

ibukota.put("Malaysia", "Kuala Lumpur");

ibukota.put("Inggris", "London");

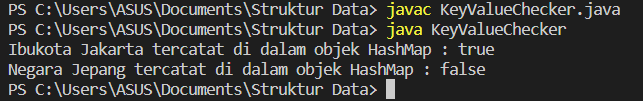
System.out.println("Ibukota Jakarta tercatat di dalam objek HashMap : " + ibukota.containsValue("Jakarta"));

System.out.println("Negara Jepang tercatat di dalam objek HashMap : "

+ ibukota.containsKey("Jepang"));

}

}

****

3. Jelaskan kegunaan method containsValue dan containsKey serta return value-nya!

Jawab:

**Latihan 16: Mendapatkan value dari suatu key di dalam objek HashMap**

import java.util.\*;

public class GetMapValue {

public static void main(String[] args) {

HashMap<String, String> ibukota = new HashMap<String, String>();

ibukota.put("Indonesia", "Jakarta");

ibukota.put("Malaysia", "Kuala Lumpur");

ibukota.put("Inggris", "London");

Set<String> daftarNegara = ibukota.keySet();

Iterator<String> negara = daftarNegara.iterator();

while (negara.hasNext()) {

String namaNegara = negara.next();

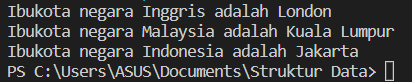
System.out.println("Ibukota negara " + namaNegara + " adalah " +

ibukota.get(namaNegara));

}

}

}

****

3. Jelaskan fungsi method get pada objek HashMap!

Jawab:

4. Apa tujuan penggunaan method keySet pada contoh di atas?

Jawab:

1. **Tugas Praktikum**

Tuliskan dan jabarkan hasil pengerjaan Tugas Praktikum yang tertera di dalam modul lengkap dengan *source code* yang digunakan.

1. Terdapat array dengan data 23, 6, 47, 35, 2, dan 14. Dengan menggunakan array tersebut, lakukan hal-hal berikut:

a. Urutkan data-data yang berada di dalam array.

import java.util.Arrays;

public class seturut {

  public static void main(String[] args) {

    int[] arrNum = {23, 6, 47, 35, 2, 16};

    Arrays.sort(arrNum);

    System.out.println("data yang urut: " + Arrays.toString(arrNum));

  }

}



b. Hitung nilai rata-rata dari data-data yang ada di dalam array tersebut.

import java.util.Scanner;

public class RataArray {

  public static void main(String[] args){

    int array[], i, n;

    array = new int[10];

    float rata, total=0;

    Scanner scan = new Scanner(System.in);

    System.out.print("Masukkan banyaknya elemen array: ");

    n = scan.nextInt();

    for(i = 1; i <= n; i++){

      System.out.print("Nilai Ke-"+ i +" : ");

      array[i] = scan.nextInt();

      total = total + array[i];

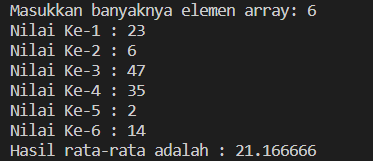
    }

    rata = total/n;

    System.out.println("Hasil rata-rata adalah : " + rata);

  }

}



c. Dapatkan data minimum dan maksimum dari array tersebut.

import java.util.\*;

public class maxmin{

    public static void main(String[] args) {

        int[] a = {

                23,

                6,

                47,

                35,

                2,

                14

        };

        System.out.println("mencari nilai  max  dan min");

        int max=a[0];

        int min=a[0];

        for (int i = 0; i < a; i++) {

            if (a[i]>max){

                max = a[i];

            }else if(a[i]<min){

                min = a[i];

            }

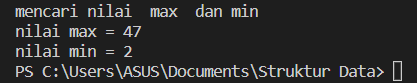
        }

        System.out.println("nilai max = "+max);

        System.out.println("nilai min = "+min);

    }

}



d. Tampilkan data-data yang termasuk ke dalam bilangan ganjil.

public class ganjil {

    public static void main(String[] args) {

        int [] bil = {23, 6, 47, 35, 2, 14};

        System.out.println("");

        System.out.print("Bilangan Ganjil :  ");

        for (int i = 0; i < bil.length; i++){

            if (bil [i] % 2!= 0){

                System.out.print(bil[i] + " ");

            }

        }

        System.out.println("");

    }

}



e. Tampilkan data-data yang termasuk ke dalam bilangan prima.

import java.util.Arrays;

import java.util.Vector;

class prima

{

    static int primeCount(int arr[], int n)  {

        int max\_val = Arrays.stream(arr).max().getAsInt();

        Boolean[] prime = new Boolean[max\_val + 1];

        for (int i = 0; i < max\_val + 1; i++)

        {

            prime[i] = true;

        }

        // Remaining part of SIEVE

        prime[0] = false;

        prime[1] = false;

        for (int p = 2; p \* p <= max\_val; p++)

        {

            if (prime[p] == true)

            {

                for (int i = p \* 2; i <= max\_val; i += p)

                {

                    prime[i] = false;

                }

            }

        }

        int count = 0;

        for (int i = 0; i < n; i++)

        {

            if (prime[arr[i]])

            {

                count++;

            }

        }

        return count;

    }

    public static void main(String[] args)

    {

        int arr[] = {23, 6, 47, 35, 2, 14};

        int n = arr.length;

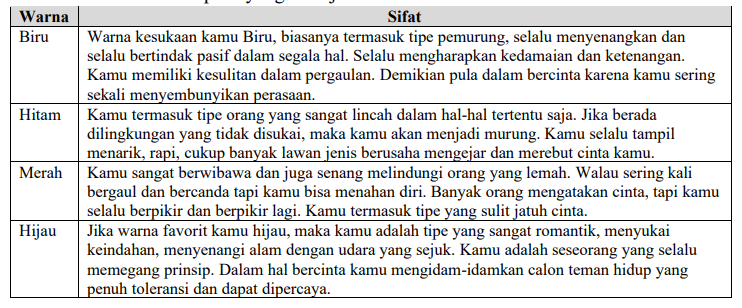
        System.out.println("Bilangan primanya adalah " + n);

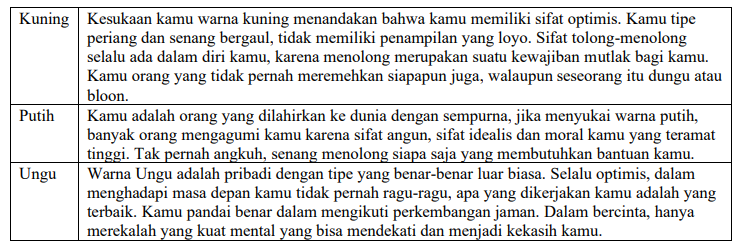
    }

}



2. Buatlah sebuah objek Map yang menyimpan data macam-macam warna beserta sifat dari warna-warna tersebut seperti yang ditunjukkan oleh tabel berikut:





Dengan memanfaatkan objek tersebut, buatlah sebuah program yang menerima input dari user berupa warna favoritnya kemudian dapat menampilkan karakter/sifat user berdasarkan warna favoritnya tersebut

import java.util.\*;

public class color{

    public static void main(String[] args) {

        Scanner userInput = new Scanner(System.in);

        String[] warna = {"Biru", "Hitam", "Merah", "Hijau", "Kuning", "Putih", "Ungu"};

        String[] sifat = {

                " Warna kesukaan kamu Biru, biasanya termasuk tipe pemurung, selalu menyenangkan dan\n" +

                "selalu bertindak pasif dalam segala hal. Selalu mengharapkan kedamaian dan ketenangan.\n" +

                "Kamu memiliki kesulitan dalam pergaulan. Demikian pula dalam bercinta karena kamu sering\n" +

                "sekali menyembunyikan perasaan.",

                "Hitam termasuk tipe orang yang sangat lincah dalam hal-hal tertentu saja. Jika berada\n" +

                "dilingkungan yang tidak disukai, maka kamu akan menjadi murung. Kamu selalu tampil\n" +

                "menarik, rapi, cukup banyak lawan jenis berusaha mengejar dan merebut cinta kamu.",

                "Merah sangat berwibawa dan juga senang melindungi orang yang lemah. Walau sering kali\n" +

                "bergaul dan bercanda tapi kamu bisa menahan diri. Banyak orang mengatakan cinta, tapi kamu\n" +

                "selalu berpikir dan berpikir lagi. Kamu termasuk tipe yang sulit jatuh cinta",

                "Jika warna favorit kamu hijau, maka kamu adalah tipe yang sangat romantik, menyukai\n" +

                "keindahan, menyenangi alam dengan udara yang sejuk. Kamu adalah seseorang yang selalu\n" +

                "memegang prinsip. Dalam hal bercinta kamu mengidam-idamkan calon teman hidup yang\n" +

                "penuh toleransi dan dapat dipercaya.",

                "Kesukaan kamu warna kuning menandakan bahwa kamu memiliki sifat optimis. Kamu tipe\n" +

                "periang dan senang bergaul, tidak memiliki penampilan yang loyo. Sifat tolong-menolong\n" +

                "selalu ada dalam diri kamu, karena menolong merupakan suatu kewajiban mutlak bagi kamu.\n" +

                "Kamu orang yang tidak pernah meremehkan siapapun juga, walaupun seseorang itu dungu atau\n" +

                "bloon.",

                "Kamu adalah orang yang dilahirkan ke dunia dengan sempurna, jika menyukai warna putih,\n" +

                "banyak orang mengagumi kamu karena sifat angun, sifat idealis dan moral kamu yang teramat\n" +

                "tinggi. Tak pernah angkuh, senang menolong siapa saja yang membutuhkan bantuan kamu",

                "Warna Ungu adalah pribadi dengan tipe yang benar-benar luar biasa. Selalu optimis, dalam\n" +

                "menghadapi masa depan kamu tidak pernah ragu-ragu, apa yang dikerjakan kamu adalah yang\n" +

                "terbaik. Kamu pandai benar dalam mengikuti perkembangan jaman. Dalam bercinta, hanya\n" +

                "merekalah yang kuat mental yang bisa mendekati dan menjadi kekasih kamu.",

        };

        System.out.print("Input Warna");

        System.out.print("Masukkan warna yang Anda sukai: ");

        String inputWarna = userInput.nextLine();

        if (inputWarna.equalsIgnoreCase("Biru")) {

            System.out.println("Karakter Anda: "+sifat[0]);

        } else if (inputWarna.equalsIgnoreCase("Hitam")) {

            System.out.print("Karakter Anda: "+ sifat[1]);

        }else if (inputWarna.equalsIgnoreCase("Merah")) {

            System.out.print("Karakter Anda: "+ sifat[2]);

        }else if (inputWarna.equalsIgnoreCase("Hijau")) {

            System.out.print("Karakter Anda: "+ sifat[3]);

        }else if (inputWarna.equalsIgnoreCase("Kuning")) {

            System.out.print("Karakter Anda: "+ sifat[4]);

        }else if (inputWarna.equalsIgnoreCase("Putih")) {

            System.out.print("Karakter Anda: "+ sifat[5]);

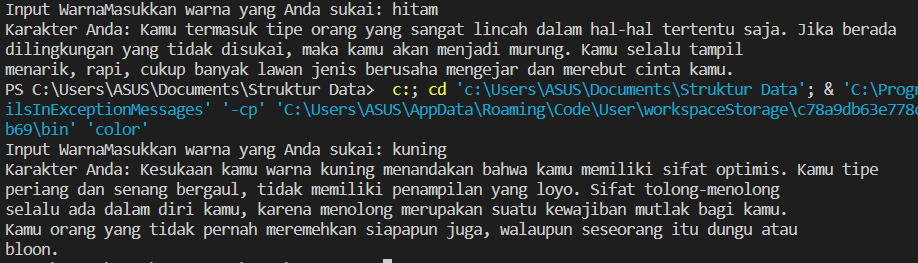
        }else if (inputWarna.equalsIgnoreCase("Ungu")) {

            System.out.print("Karakter Anda: "+ sifat[5]);

        }

    }

}



3. Terdapat dua buah himpunan sebagai berikut:

𝐴 = {1,2,3,4,5}

𝐵 = {5,6,7,8,9,10}

Dengan memanfaatkan HashSet dapatkan output dari operasi himpunan berikut:

a. 𝐴 – 𝐵

b. 𝐴 ∩ 𝐵

c. 𝐴 ∪ 𝐵

import java.util.\*;

public class Himpunan {

    public static void main(String[] args) {

        Set s1 = new HashSet();

        s1.add("1");

        s1.add("2");

        s1.add("3");

        s1.add("4");

        s1.add("5");

        Set s2 = new HashSet();

        s2.add("5");

        s2.add("6");

        s2.add("7");

        s2.add("8");

        s2.add("9");

        s2.add("10");

        Set union = new TreeSet(s2);

        union.addAll(s1); //untuk menggabungkan

        print("Gabungan", union);

        Set intersect = new TreeSet(s1);

        intersect.retainAll(s2); //untuk interseksi

        print("irisan",intersect);

        Set min = new TreeSet(s1);

        min.removeAll(s2);

        print("min",min);

        Set himpunanBagian = new TreeSet(s1);

        himpunanBagian.containsAll(s2); //himpunan bagian

        print("Bagian",himpunanBagian);

    }

    protected static void print(String label,Collection c){

        System.out.println("--------------" + label + "--------------");

        Iterator it = c.iterator();

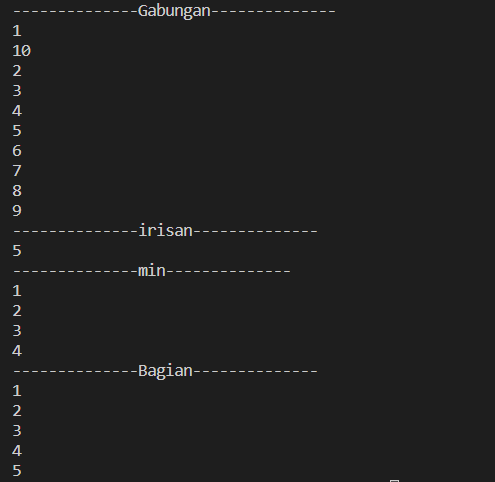
        while(it.hasNext()){

            System.out.println(it.next());

        }

    }

}



d. 𝐴 ⊂ B