[注意]: 有關HashSet的contains()函數:

1. 會先呼叫hashCode()以確定該物件「可能」存在HashSet中。

2. 通過後，仍利用equals()函數來進行比對，以確認其存在。

--> 因此，第一題請同時覆寫hashCode()與equals()。

1. [Constructor+HashCode+toString]: 完成以下程式，以產生規定功能。

[程式片段]:

import java.util.\*;

class Student { //假設每個欄位都有意義

int id; String name; double[] gpa;

Student() { }

Student(int a, String b, double[] c) {

// DIY here

}

public int hashCode() { // DIY here

int code = 17 ;

// DIY here

return code ;

}

public String toString() {

// DIY, 印出的格式自行設計，需印出gpa[]中的每個成績

return "("+id+","+name+","+gpa+")";

}

}

public class HWK5\_XX\_YY\_01 {

static HashSet<Student> hs = new HashSet<Student>();

public static void main(String[] args) {

buildHashSet() ;

Student query = new Student(1001, "Peter", new double[]{4.1,3.9,3.3}) ; ;

if (hs.contains(query)) {

System.out.println("Yes, hs contains "+query); // 應該印出這一行

} else {

System.out.println("No, "+query+" is not found in hs.");

}

}

public static void buildHashSet() {

hs.add(new Student(1001, "Peter", new double[]{4.1,3.9,3.3})) ;

hs.add(new Student(1002, "Mary", new double[]{4.2, 4.0, 4.2})) ;

hs.add(new Student(1003, "Paul", new double[] {3.7, 3.5, 3.3, 4.0})) ;

System.out.println("Hash Set=" + hs);

}

}

2.[Copy constructor]: 依照說明完成以下程式，產生規定輸出。

[程式片段]:

import java.util.\* ;

class Number {

int x ;

Number() { }

Number(int a) {x=a; }

Number(Number n) {this.x=n.x;}

}

class MyScore2 {

// --- 將元素加入LinkedList中請用add() --

private LinkedList<Number> scoreList = new LinkedList<Number>() ;

private int rank = -1 ;

MyScore2(int[] score, int rank) { // conversion constructor

// DIY: deep copy

}

MyScore2(MyScore2 ms2) { // copy constructor

// DIY: deep copy

}

public void clear() {

scoreList.clear(); rank=-1;

}

public String toString() {

// DIY: 格式請參考main()中的指示

return "Not Yet" ;

}

}

public class public class HWK5\_XX\_YY\_02 {

public static void main(String[] args){

int[] ss = {77, 99, 88, 66} ;

MyScore2 ms1 = new MyScore2(ss, 8) ;

System.out.println("ms1="+ms1) ; // 印出[77,99,88,66] (r:8)

MyScore2 ms2 = new MyScore2(ms1) ;

ms1.clear() ; // 本行不可刪除

System.out.println("ms2="+ms2) ; // 印出[77,99,88,66] (r:8)

}

}