// Java program to illustrate the concept

// of Autoboxing and Unboxing

import java.io.\*;

class boxing

{

public static void main (String[] args)

{

// creating an Integer Object

// with value 10.

Integer i = new Integer(10);

// unboxing the Object

int i1 = i;

System.out.println("Value of i: " + i);

System.out.println("Value of i1: " + i1);

//Autoboxing of char

Character gfg = 'a';

// Auto-unboxing of Character

char ch = gfg;

System.out.println("Value of ch: " + ch);

System.out.println("Value of gfg: " + gfg);

}

}

import java.io.\*;

class exception

{

public static void main(String args[])throws IOException

{

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

int a=Integer.parseInt(br.readLine());

int b=Integer.parseInt(br.readLine());

try

{

int d = a / b;

}

catch(Exception e)

{

System.out.println(e);

}

}

}

import java.io.\*;

import java.util.\*;

class MyThread extends Thread{

Thread t;

String name;

MyThread(String n)

{

name=n;

t=new Thread(n);

}

public void run(){

System.out.println("thread is running...");

//t.sleep(1000);

for (int i=10;i!=0;i--)

System.out.println(name +" " + i);

try{

t.sleep(10);

}

catch (Exception e){

System.out.println(e);

} } }

class Multithread{

public static void main(String args[]){

MyThread t1 = new MyThread("Ankur");

MyThread t2 = new MyThread("Debomoy");

t1.start();

t2.start();

}

}

import java.io.\*;

import java.util.\*;

class MythreadImp implements Runnable{

String name;

MythreadImp(String n)

{

name=n;

}

public void run(){

System.out.println("thread is running...");

//t.sleep(1000);

for (int i=10;i!=0;i--)

System.out.println(name +" " + i);

}

}

class MultiThreadImp{

public static void main (String args[])

{

MythreadImp m1 = new MythreadImp("Ankur");

MythreadImp m2 = new MythreadImp("Debomoy");

Thread t1 =new Thread(m1);

Thread t2 =new Thread(m2);

t1.start();

t2.start();

t1.setPriority(8);

try{

t1.join(2000);

}

catch(Exception e){

System.out.println(e);

}

System.out.println("Get priority "+t1.getPriority());

System.out.println("Get priority "+t2.getPriority());

try{

t1.sleep(10);

}

catch (Exception e){

System.out.println(e);

}

System.out.println("Is alive "+t2.isAlive());

}

}

import java.io.\*;

class Student

{

int rollno; String name; String dept; int marks;

void get(int r, String n, String d, int m)

{rollno = r; name = n; dept = d; marks = m;}

static void findHighest(Student arr[])

{

int max = 0; String max\_name="";

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

{

if(arr[i].marks > max)

{

max = arr[i].marks;

max\_name = arr[i].name;

}

}

System.out.println("The student who got highest marks= "+ max\_name);

}

static void findLowest(Student arr[])

{

long min = 9999999; String min\_name=" ";

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

{

if(arr[i].marks < min)

{

min = arr[i].marks;

min\_name = arr[i].name;

}

}

System.out.println("The student who got lowest marks= "+ min\_name);

}

static void findMoreThanAvg(Student arr[])

{

double average, sum=0;

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

{

sum = sum + arr[i].marks;

}

average = sum / (arr.length);

System.out.println("The students who got more than average marks: ");

for(int j=0; j<arr.length; j++)

{

if(arr[j].marks > average)

{

System.out.println(arr[j].name);

} } }

public static void main(String args[])throws IOException

{

Student ob[] = new Student[5];

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

for (int i=0; i<5; i++){

ob[i] = new Student();

System.out.println("Enter roll no, name, department and marks= ");

int a = Integer.parseInt(br.readLine());

String b = br.readLine();

String c = br.readLine();

int d = Integer.parseInt(br.readLine());

ob[i].get(a,b,c,d);}

findHighest(ob);

findLowest(ob);

findMoreThanAvg(ob); } }