



1. Question 1

ALL



1



Create a class **PhotoFrame** with below attributes:

id - int

length - int

breadth - int

material - String

Where **id** is the id of the photo frame, **length** and **breadth** are the length and breadth of the photo frame(in cm) and **material** is the material used for making the photo frame.

The above attributes should be private, write getters, setters and parameterized constructor as required.

Create class **Solution** with main method. Implement two static methods in Solution class-

getAverageAreaOfPhotoFramesByMaterial and

findPhotoFrameWithSecondLargestBreadth.

hackerrank.com/test/5pasm2gd342/questions/32190e3f7

10m left

I and
findPhotoFrameWithSecondLargestBreadth.



ALL



1



getAverageAreaOfPhotoFramesByMaterial
method:
The method will take array of PhotoFrame objects and one String value(material of photo frame) as input parameters and the method return the average area of the PhotoFrames made of that material available in the array of PhotoFrame objects.
If the average area is less than 25 then return 0 else return the average area.

Equation for calculating area of a photo frame(rectangle): area = length * breadth
Note :search should be case insensitive

findPhotoFrameWithSecondLargestBreadth
method:
This method will take an array of PhotoPhrame objects as input and find the photo frame object having the second largest breadth.

If the length and breadth of the second largest object is less than 5 then return null else return the object.

Note : No two PhotoFrame objects will

 HackerRank

 hackerrank.com/test/5pasm2gd342/questions/52153

10m left



ALL



1



If the length and breadth of the second largest object is less than 5 then return null else return the object.

Note : No two PhotoFrame objects will have the same breadth.

The above mentioned static methods should be called from the main method.

For `getAverageAreaOfPhotoFramesByMaterial` method - The main method should print the returned average Area as it is if the returned value is greater than 0 or it should print "No such Photo Frame available."

For `findPhotoFrameWithSecondLargestBreadth` method - The main method should print the id,length,breadth and material in separate lines from the returned PhotoFrame object if the returned value is not null.
If the returned value is null then it should print "No such Photo Frame available."

Before calling these static methods in main, use Scanner object to read the length of the array (number of objects need to be added into the array) and then add photo frame objects into the array referring attributes in the above mentioned attribute sequence.

9m left



ALL



1



array (number of objects need to be added into the array) and then add photo frame objects into the array referring attributes in the above-mentioned attribute sequence.

Next, read the material name as string input for searching and finding the average area of photo frames made of that material.

Consider below sample input and output:

Input1:

4

101

15

15

wood

102

16

16

wood

103

20

30

plastic

104

14

14

plastic

wood

9m left

plastic

wood



ALL



Output:

240.5

Id-102

Length-16

Breadth-16

Material-wood

1

Input2:

4

501

1

2

✓

wood

502

5

3

wood

503

3

4

plastic

504

1

5

plastic

9m left

503

3

4

plastic

504

1

5

plastic

wood



Output:

No such Photo Frame available.

No such Photo Frame available.

1



Sample code snippet for reference:

Please use below code to build your solution.

```
import java.util.Scanner;
public class Solution
{
    public static void main(String[] args)
    {
        //code to read values
        //code to call required method
        //code to display the result
    }
    //code the first method
}
```

```
1 import java.io.*;
2 import java.util.*;
3 import java.text.*;
4 import java.math.*;
5 import java.util.regex.*;
6
7 public class Solution {
8     public static void main(String args[] ) throws Exception {
9         Scanner sc = new Scanner(System.in);
10        int n = sc.nextInt(); sc.nextLine();
11        PhotoFrame [] arr = new PhotoFrame[n];
12        for(int i=0; i< arr.length; i++){
13            int a = sc.nextInt(); sc.nextLine();
14            int b = sc.nextInt(); sc.nextLine();
15            int c = sc.nextInt(); sc.nextLine();
16            String d = sc.nextLine();
17            arr[i] = new PhotoFrame(a,b,c,d);
18        }
19        String input = sc.nextLine();
20        sc.close();
21
22        double obj1 = getAverageOfPhoto(arr, input);
23        if(obj1 > 25){
24            System.out.println(obj1);
25        }
    }
```

Line: 23

Test Results

Q. Search



Custom Input

Run Code

Run Tests

Sub

ENG IN 08:21 F 13-05-2024

```
double obj1 = getAverageOfPhoto(arr, input);
if(obj1 > 25){
    System.out.println(obj1);
}else{
    System.out.println("No such Photo Frame available.");
}

int obj2 = findPhotoSecondLargestBreadth(arr);
if(obj2 > 0){
    for(int j =0; j <arr.length; j++){
        if(arr[j].breadth == obj2){
            System.out.println("Id-"+arr[j].id);
            System.out.println("Length-"+arr[j].length);
            System.out.println("Breadth-"+arr[j].breadth);
            System.out.println("Material-"+arr[j].material);
        }
    }
}else{
    System.out.println("No such Photo Frame available.");
}

public static double getAverageOfPhoto(PhotoFrame[] arr, String input){
    int area = 0;
    int count = 0;
```

```
}
```

```
public static double getAverageOfPhoto(PhotoFrame[] arr, String input){
```

```
    int area = 0;
```

```
    int count = 0;
```

```
    double sum = 0;
```

```
    double average = 0;
```

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

```
        if(arr[i].material.equalsIgnoreCase(input)){
```

```
            area = arr[i].length * arr[i].breadth;
```

```
            sum += area;
```

```
            count ++;
```

```
        }
```

```
}
```

```
    if(count > 0){
```

```
        average = sum / count;
```

```
        return average;
```

```
    }else{
```

```
        return 0;
```

```
}
```

```
public static int findPhotoSecondLargestBreadth(PhotoFrame[] arr){
```

```
    int[] refined = new int[0];
```

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

Line: 23 Col: 21

Results

Custom Input

Run Code

Run Tests

19 May 2024
Submit



ENG IN 08:21 PM 13-05-2024



```
public static int findPhotoSecondLargestBreadth(PhotoFrame[] arr){  
    int[] refined = new int[0];  
    for (int i = 0; i < arr.length; i++) {  
        if(arr[i].length > 5 && arr[i].breadth > 5){  
            refined = Arrays.copyOf(refined, refined.length + 1);  
            refined[refined.length - 1] = arr[i].breadth;  
        }  
    }  
    Arrays.sort(refined);  
    if(refined.length > 1){  
        return refined[refined.length - 2];  
    }else{  
        return 0;  
    }  
}  
}  
class PhotoFrame{  
    int id;  
    int length;  
    int breadth;  
    String material;  
    public PhotoFrame(int id, int length, int breadth, String material){  
        this.id=id;  
        this.length=length;  
        this.breadth=breadth.  
    }  
}
```

Line: 23 Col: 21

Network 60
Internet access

```
class PhotoFrame{  
    int id;  
    int length;  
    int breadth;  
    String material;  
    public PhotoFrame(int id, int length, int breadth, String material){  
        this.id=id;  
        this.length=length;  
        this.breadth=breadth;  
        this.material=material;  
    }  
}
```

Write a java program to check if two given numbers are amicable. The two numbers will be given as input.

Amicable numbers are a pair of two different numbers such that the sum of the proper divisors of each number in the pair is equal to the other number.

Print "The pair is amicable" if they are amicable and "The pair is not amicable" if they are not amicable.

Sample Test Cases

Input:

220

284

Output:

The pair is amicable

Input2:

50

20

Output:

The pair is not amicable

Language: Java 15

Environment

Autocomplete Ready

```
7 ✓ public class Solution {  
8 ✓     public static int sumDivisor(int n){  
9         int sum = 1;  
10    ✓    for(int i = 2; i*i <= n; i++){  
11    ✓        if(n % i == 0){  
12            sum += i;  
13            if(i != n/i){  
14                sum += n/i;  
15            }  
16        }  
17    }  
18    return sum;  
19 }  
20 ✓ public static boolean areTheseAmicable(int n1, int n2){  
21    return sumDivisor(n1) == n2 && sumDivisor(n2) == n1;  
22 }  
23 ✓ public static void main(String args[]){  
24    Scanner sc = new Scanner(System.in);  
25    int n1 = sc.nextInt();  
26    int n2 = sc.nextInt();  
27  
28    ✓ if(areTheseAmicable(n1, n2)){  
29        System.out.println("The pair is amicable");  
30    }else{  
31    }  
32 }
```

Line: 7

Test Results

Search

Custom Input

Run Code

Run Tests

Sub

ENG IN 08:22 PM
13-05-2024

ASUS

```
if(n % i == 0){
    sum += i;
    if(i != n/i){
        sum += n/i;
    }
}
return sum;
}

public static boolean areTheseAmicable(int n1, int n2){
    return sumDivisor(n1) == n2 && sumDivisor(n2) == n1;
}

public static void main(String args[]) {
    Scanner sc = new Scanner(System.in);
    int n1 = sc.nextInt();
    int n2 = sc.nextInt();

    if(areTheseAmicable(n1, n2)){
        System.out.println("The pair is amicable");
    }else{
        System.out.println("The pair is not amicable");
    }
}
```