Sorting: Comparator

by Shafaet

Check out the resources on the page's right side to learn more about sorting. The video tutorial is by Gayle Laakmann McDowell, author of the best-selling interview book Cracking the Coding Interview.

Comparators are used to compare two objects. In this challenge, you'll create a comparator and use it to sort an array. The *Player* class is provided in the editor below; it has two fields:

1. A string, $name$.
2. An integer, $score$.

Given an array of n *Player* objects, write a comparator that sorts them in order of decreasing score; if 2 or more players have the same score, sort those players alphabetically by name. To do this, you must create a *Checker* class that implements the *Comparator* interface, then write an *int compare(Player a, Player b)* method implementing the Comparator.compare(T o1, T o2) method.

Input Format

Locked stub code in the *Solution* class handles the following input from stdin:
The first line contains an integer, n, denoting the number of players.
Each of the n subsequent lines contains a player's respective $name$ and $score$.

Constraints

- $0 \leq score \leq 1000$
- Two or more players can have the same name.
- Player names consist of lowercase English alphabetic letters.

Output Format

You are not responsible for printing any output to stdout. Locked stub code in *Solution* will create a *Checker* object, use it to sort the *Player* array, and print each sorted element.

Sample Input

```
5
amy 100
david 100
heraldo 50
aakansha 75
aleksa 150
```

Sample Output

```
aleksa 150
amy 100
david 100
aakansha 75
heraldo 50
```

Explanation

As you can see, the players are first sorted by decreasing score and then sorted alphabetically by name.

Submissions: 4793
Max Score: 35

Difficulty: Medium

Rate This Challenge:

Need Help?

8:54

QuickSort

More

Current Buffer (saved locally, editable)

Java 7

```
import java.util.*;
// Write your Checker class here

class Checker implements Comparator<Player> {
    @Override
    public int compare(Player a, Player b) {
        int diff = b.score - a.score;
        return diff != 0 ? diff : a.name.compareTo(b.name);
    }
}
class Player{
    String name;
    int score;

    Player(String name, int score){
        this.name = name;
        this.score = score;
    }
}

class Solution {

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

        Player[] player = new Player[n];
        Checker checker = new Checker();

        for(int i = 0; i < n; i++){
            player[i] = new Player(scan.next(), scan.nextInt());
        }
        scan.close();


        Arrays.sort(player, checker);
        for(int i = 0; i < player.length; i++){
            System.out.printf("%s %s\n", player[i].name, player[i].score);
        }
    }
}
```

Line: 10 Col: 2

Upload Code as File Test against custom input Run Code Submit Code

Congrats, you solved this challenge!

- ✔ Test Case #0
- ✔ Test Case #1
- ✔ Test Case #2
- ✔ Test Case #3
- ✔ Test Case #4
- ✔ Test Case #5

Next Challenge