Combinations

Given two integers n and k, return all possible combinations of k numbers out of 1 2 3 ... n.

Make sure the combinations are **sorted**.

To elaborate,

1. Within every entry, elements should be sorted. [1, 4] is a valid entry while [4, 1] is not.
2. Entries should be sorted within themselves.

public class Solution **{**

private int k **=**0**,**n**=**0**; //Global variables**

private ArrayList**<**ArrayList**<**Integer**>>** ListArr **=** **new** ArrayList**<**ArrayList**<**Integer**>>();**

public ArrayList**<**ArrayList**<**Integer**>>** combine**(**int A**,** int B**)** **{**

n **=** A**;**

k **=** B**;**

combinations**(**1**,new** ArrayList**<>());**

**return** ListArr**;**

**}**

public void combinations**(**int index**,**ArrayList**<**Integer**>** set**)** **{**

**if(**set**.**size**()** **==** k**){**

ListArr**.**add**(new** ArrayList**(**set**));**

**return;**

**}**

**for(**int i **=** index**;**i**<=**n**;**i**++){**

set**.**add**(**i**);**

combinations**(**i**+**1**,** set**);**

set**.**remove**(**set**.**size**()-**1**);//Make the backtraking**

**}**

**}**

**}**