768. Yang Hui Triangle

* [Description](http://lintcode.com/en/problem/yang-hui-triangle/" \l "description)
* [Notes](http://lintcode.com/en/problem/yang-hui-triangle/#note)
* [Testcase](http://lintcode.com/en/problem/yang-hui-triangle/#testcase)
* [Judge](http://lintcode.com/en/problem/yang-hui-triangle/#judge)

Given an integer n, return the first n-lineYang Hui triangle.

 Notice

1.0<=n<=20  
2.Yang Hui’s Triangle also called Pascal's triangle. --([Wikipedia](https://en.wikipedia.org/wiki/Pascal%27s_triangle" \o "Wikipedia))

Have you met this question in a real interview?

Yes

**Example**

Given n = 4

return

[

[1]

[1,1]

[1,2,1]

[1,3,3,1]

]

<http://lintcode.com/en/problem/yang-hui-triangle/#>

<https://www.geeksforgeeks.org/pascal-triangle/>

/\*

\* To change this license header, choose License Headers in Project Properties.

\* To change this template file, choose Tools | Templates

\* and open the template in the editor.

\*/

package javaapplication3;

import java.util.ArrayList;

import java.util.List;

/\*\*

\*

\* @author Usuario

\*/

public class JavaApplication3 {

/\*\*

\* @param n

\* @param args the command line arguments

\*/

public List<List<Integer>> calcYangHuisTriangle(int n) {

// write your code here

int[][] arr = new int[n][n]; // An auxiliary array to store generated pscal triangle values

List<List<Integer>> matriz =

new ArrayList();

// Iterate through every line and print integer(s) in it

for (int line = 0; line < n; line++)

{

List<Integer> fila = new ArrayList<Integer>();

// Every line has number of integers equal to line number

for (int i = 0; i <= line; i++)

{

// First and last values in every row are 1

if (line == i || i == 0)

arr[line][i] = 1;

else // Other values are sum of values just above and left of above

arr[line][i] = arr[line - 1][i - 1] + arr[line - 1][i];

//Console.Write(arr[line, i] + " ");

fila.add(arr[line][i]);

}

//printf("\n");

//Console.WriteLine();

matriz.add(fila);

}

return matriz;

}

public static void main(String[] args) {

// TODO code application logic here

}

}