Pascal's Triangle

Link:-

<https://leetcode.com/problems/pascals-triangle/description/>

Given an integer numRows, return the first numRows of **Pascal's triangle**.

In **Pascal's triangle**, each number is the sum of the two numbers directly above it as shown:



**Example 1:**

**Input:** numRows = 5

**Output:** [[1],[1,1],[1,2,1],[1,3,3,1],[1,4,6,4,1]]

Solution:-

import java.util.\*;

class Solution {

public List<List<Integer>> generate(int numRows) {

List<List<Integer> > pascal

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

if(numRows>0){

pascal.add(new ArrayList<>());

pascal.get(0).add(1);

if(numRows==1)

return pascal;

pascal.add(new ArrayList<>());

pascal.get(1).add(1);

pascal.get(1).add(1);

if(numRows==2)

return pascal;

for(int i=2;i<numRows;i++){

pascal.add(new ArrayList<>());

pascal.get(i).add(1);

for(int j=0;j<pascal.get(i-1).size()-1;j++){

pascal.get(i).add(pascal.get(i-1).get(j)+pascal.get(i-1).get(j+1));

}

pascal.get(i).add(1);

}

return pascal;

}

return pascal;

}

}