ARRAYS | LEETCODE 118 |

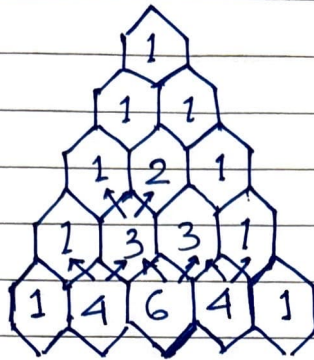## PASCAL'S Triangle

Approach

1) Create List inside a List (with size = row No)

2) Each sublist size = row No so iteration should start from 1 to numRows
↓

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

for(int i=1; i<=numRows; i++) {
    pascalList.add (new ArrayList <Integer> (i));
}
↑
current
row No.

3) First & Last element of each sublist =1
Create a 2nd forloop from 0 to i as below

↓

for (int j= 0; j< i; j++) {
    if (j==0 || j == i-1) { // first & last element)
        pascalList.get (i-1). add (1);

4) current
Row index
= Sum(prev Row
index-1)
} else {   (pL)
    prevRow's Left = pascalList.get (i-2).get(j-1);
    prevRow Right = ... (i-2).get(j);
        (pR)
+ prev Row
index)
    pascalList.get(i-1). add (pL+ pR);
}

classmate }
  }