Solution 1 Solution 2

30 } 31 Run Code

Our Solution(s)

Run Code

**Your Solutions** 

12px

Solution 1 Solution 2 Solution 3

```
_{\rm 1} // Copyright @ 2020 AlgoExpert, LLC. All rights reserved.
   import java.util.*;
   class Program {
      // O(n*n!) time | O(n*n!) space
      public static List<List<Integer>> getPermutations(List<Integer> array) {
       List<List<Integer>> permutations = new ArrayList<List<Integer>>();
        getPermutations(0, array, permutations);
        return permutations;
12
13
      public static void getPermutations(int i, List<Integer> array, List<List<Integer</pre>
14
       if (i == array.size() - 1) {
         permutations.add(new ArrayList<Integer>(array));
16
        } else {
         for (int j = i; j < array.size(); j++) {</pre>
18
           swap(array, i, j);
           getPermutations(i + 1, array, permutations);
19
20
            swap(array, i, j);
22
       }
23
24
25
      public static void swap(List<Integer> array, int i, int j) {
26
       Integer tmp = array.get(i);
27
        array.set(i, array.get(j));
28
        array.set(j, tmp);
29
```

```
import java.util.*;

class Program {
  public static List<List<Integer>> getPermutations(List<Integer> array) {
    // Write your code here.
    return null;
  }
}
```

Custom Output Raw Output Submit Code

Run or submit code when you're ready.