77. Combinations

Description | Hints | Submissions | Discuss | Solution

Pick One

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

Example:

```
Input: n = 4, k = 2
Output:
[
  [2,4],
  [3,4],
  [2,3],
  [1,2],
  [1,3],
  [1,4],
]
```

```java
/*
 * 这是个回溯算法。用一个循环递归处理子问题，问题的终止条件是每个组合中的
 * 元素个数达到k个
 */
public class L77 {

    public List<List<Integer>> combine(int n, int k) {
        List<List<Integer>> res = new ArrayList<List<Integer>>();
        List<Integer> temp = new ArrayList<Integer>();
        dfs(res, temp, n, k, 1);
        return res;
    }

    public void dfs(List<List<Integer>> res, List<Integer> temp, int n, int k, int m) {
        if(k == 0)
        {
            res.add(new ArrayList<Integer>(temp));
            return;
        }
        for(int i = m; i <= n; i++) {
            temp.add(i);
            dfs(res, temp, n, k-1, i+1);
            temp.remove(temp.size() - 1);
        }
    }

}
```