Subsets

Question: Given a set of distinct integers, nums, return all possible subsets.

Note: The solution set must not contain duplicate subsets.

For example: If nums = [1,2,3], a solution is:

[[3], [1], [2], [1,2,3], [1,3], [2,3], [1,2], []].

Solutions:

```
class Solution:
    # @param {integer[]} nums
    # @return {integer[][]}
    def subsets(self, nums):
        if nums is None:
            return []
        result = []
        nums.sort()
        self.dfs(nums, 0, [], result)
        return result

    def dfs(self, nums, pos, list_temp, ret):
        # append new object with []
        ret.append([] + list_temp)
        for i in range(pos, len(nums)):
            list_temp.append(nums[i])
            self.dfs(nums, i + 1, list_temp, ret)
            list_temp.pop()

Solution().subsets([1,2,3])
```