

# 347. Top K Frequent Elements

🕒 Created	@April 10, 2022 4:33 PM
▼ Difficulty	Medium
☰ LC Url	<a href="https://leetcode.com/problems/top-k-frequent-elements/">https://leetcode.com/problems/top-k-frequent-elements/</a>
▼ Importance	
☰ Tag	Array&Sorting NEET Queue
☰ Video	<a href="https://maxming0.github.io/2020/07/17/Top-K-Frequent-Elements/">https://maxming0.github.io/2020/07/17/Top-K-Frequent-Elements/</a>

Given an integer array `nums` and an integer `k`, return the `k` most frequent elements. You may return the answer in **any order**.

## Example 1:

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

## Example 2:

```
Input: nums = [1], k = 1
Output: [1]
```

## Constraints:

- `1 <= nums.length <= 105`
- `k` is in the range `[1, the number of unique elements in the array]`.
- It is **guaranteed** that the answer is **unique**.

**Follow up:** Your algorithm's time complexity must be better than  `$O(n \log n)$` , where `n` is the array's size.


## Solution

```
class Solution:
    def topKFrequent(self, nums: List[int], k: int) -> List[int]:
        # ct = Counter(nums)
        # return heapq.nlargest(k, ct.keys(), key=ct.get)

        q = []
        for num, freq in Counter(nums).items():
            if len(q) == k:
                heapq.heappushpop(q, (freq, num))
            else:
                heapq.heappush(q, (freq, num))
        return [x[1] for x in q]
```

### Python应用--优先队列与heapq

本文始发于个人公众号：TechFlow，求个关注 今天的文章来介绍 Python当中一个蛮有用的库-- heapq。heapq的全写是heap queue，是堆队列的意思。这里的 堆和队列 都是数据结构，在后序

 <https://zhuanlan.zhihu.com/p/106170247>

