**Problem #2248: Intersection of Multiple arrays**

<https://leetcode.com/problems/intersection-of-multiple-arrays/description/>

**My Solution:**

Python3

**Approach**

1. Initialize mydict to an empty dictionary.
2. Iterate through each list in nums and get the frequency of each number in each of the lists in the dictionary.
3. Initialize res to an empty list.
4. Iterate through mydict items and get each key and value pair. If the value is equal to the length of nums, then the key is present in all the arrays in nums. This is because the elements in each list are distinct. Append the key to res.
5. Return res sorted in ascending order.

**Complexity**

* Time complexity:  
  O(n) since we only iterate through nums once.
* Space complexity:  
  O(n) since we use a dictionary to hold all the numbers of nums.

**Code:**

from collections import Counter

class Solution:

def intersection(self, nums: List[List[int]]) -> List[int]:

mydict = {}

for alist in nums:

for num in alist:

if num not in mydict.keys():

mydict[num] = 1

else:

mydict[num] += 1

res = []

for key, value in mydict.items():

if value == len(nums):

res.append(key)

return sorted(res)