**DAY-5 TASK**

Problem-1

Given an array of strings words and an integer k, return *the*k*most frequent strings*.

Return the answer **sorted** by **the frequency** from highest to lowest. Sort the words with the same frequency by their **lexicographical order**.

**Example 1:**

**Input:** words = ["i","love","leetcode","i","love","coding"], k = 2

**Output:** ["i","love"]

**Explanation:** "i" and "love" are the two most frequent words.

Note that "i" comes before "love" due to a lower alphabetical order.

**Example 2:**

**Input:** words = ["the","day","is","sunny","the","the","the","sunny","is","is"], k = 4

**Output:** ["the","is","sunny","day"]

**Explanation:** "the", "is", "sunny" and "day" are the four most frequent words, with the number of occurrence being 4, 3, 2 and 1 respectively.

**Constraints:**

* 1 <= words.length <= 500
* 1 <= words[i].length <= 10
* words[i] consists of lowercase English letters.
* k is in the range [1, The number of **unique** words[i]]

Problem-2

You are given a sorted integer array arr containing 1 and **prime** numbers, where all the integers of arr are unique. You are also given an integer k.

For every i and j where 0 <= i < j < arr.length, we consider the fraction arr[i] / arr[j].

Return *the* kth *smallest fraction considered*. Return your answer as an array of integers of size 2, where answer[0] == arr[i] and answer[1] == arr[j].

**Example 1:**

**Input:** arr = [1,2,3,5], k = 3

**Output:** [2,5]

**Explanation:** The fractions to be considered in sorted order are:

1/5, 1/3, 2/5, 1/2, 3/5, and 2/3.

The third fraction is 2/5.

**Example 2:**

**Input:** arr = [1,7], k = 1

**Output:** [1,7]

**Constraints:**

* 2 <= arr.length <= 1000
* 1 <= arr[i] <= 3 \* 104
* arr[0] == 1
* arr[i] is a **prime** number for i > 0.
* All the numbers of arr are **unique** and sorted in **strictly increasing** order.
* 1 <= k <= arr.length \* (arr.length - 1) / 2