Problem List

Description | Accepted | Editorial | Solutions | Submissions

3662. Filter Characters by Frequency

Solved

Premium

Easy | Topics | Hint

You are given a string `s` consisting of lowercase English letters and an integer `k`.

Your task is to construct a new string that contains only those characters from `s` which appear **fewer** than `k` times in the entire string. The order of characters in the new string must be the **same** as their **order** in `s`.

Return the resulting string. If no characters qualify, return an empty string.

Note: **Every occurrence** of a character that occurs fewer than `k` times is kept.

Example 1:

Input: `s = "aadbbcccca", k = 3`

Output: `"dbb"`

Explanation:

Character frequencies in `s`:

- `'a'` appears 3 times
- `'d'` appears 1 time
- `'b'` appears 2 times
- `'c'` appears 4 times

Only `'d'` and `'b'` appear fewer than 3 times. Preserving their order, the result is `"dbb"`.

Example 2:

Input: `s = "xyz", k = 2`

Output: `"xyz"`

Explanation:

All characters (`'x'`, `'y'`, `'z'`) appear exactly once, which is fewer than 2. Thus the whole string is returned.

Constraints:

- `1 <= s.length <= 100`
- `s` consists of lowercase English letters.
- `1 <= k <= s.length`

Seen this question in a real interview before? 1/5

Yes No

Accepted 973/1.1K | Acceptance Rate 87.1%

Topics

Hint 1

Discussion (2)

7 2 0 Online

Submit

Code

Python3 • Auto

```python
class Solution:
    def filterCharacters(self, s: str, k: int) -> str:

        dictx = {}

        for word in s:
            if word not in dictx.keys():
                dictx[word] = 1
            else:
                dictx[word] += 1


        res = ""

        for word in s:
            if dictx[word] < k:
                res += word

        return res
```

Saved Ln 11, Col 1

Testcase | Test Result

Accepted Runtime: 0 ms

Case 1 Case 2

Input

s =
"aadbbcccca"

k =
3

Output