

387. First Unique Character in a String

Easy 3271 158 Add to List Share

Given a string `s`, find the first non-repeating character in it and return its index. If it does not exist, return `-1`.

Example 1:

Input: `s = "leetcode"`
Output: `0`

Example 2:

Input: `s = "loveleetcode"`
Output: `2`

Example 3:

Input: `s = "aabb"`
Output: `-1`

Constraints:

- `1 <= s.length <= 105`
- `s` consists of only lowercase English letters.

Accepted 787,284 | Submissions 1,438,645

Seen this question in a real interview before?

Yes

No

```
1 class Solution:
2     def firstUniqChar(self, s: str) -> int:
3         """
4         :type s: str
5         :rtype: int
6         """
7         # build hash map : character and how often it appears
8         count = collections.Counter(s)
9
10        # find the index
11        for idx, ch in enumerate(s):
12            if count[ch] == 1:
13                return idx
14        return -1
15
16
```

Testcase

Run Code Result

Debugger

Accepted

Runtime: 70 ms

Your input

"leetcode"

Output

0

Expected

0

Diff