

First Unique Character in a String

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: a. $1 \leq s.length \leq 10^5$ b. `s` consists of only lowercase English letters.

In [12]:

```
1 s = "leetcode"
2 def firstUniqChar(s):
3     char_count = {}
4
5     for char in s:
6         char_count[char] = char_count.get(char,0)+1
7     for i in range(len(s)):
8         if char_count[s[i]]== 1:
9             return i
10    return -1
11
12 result = firstUniqChar(s)
13 print(result)
```

0

In [13]:

```
1 s = "loveleetcode"
2 def firstUniqChar(s):
3     char_count = {}
4
5     for char in s:
6         char_count[char] = char_count.get(char,0)+1
7     for i in range(len(s)):
8         if char_count[s[i]]== 1:
9             return i
10    return -1
11
12 result = firstUniqChar(s)
13 print(result)
```

2

In [14]:

```
1 s = "aabb"
2 def firstUniqChar(s):
3     char_count = {}
4
5     for char in s:
6         char_count[char] = char_count.get(char,0)+1
7     for i in range(len(s)):
8         if char_count[s[i]]== 1:
9             return i
10    return -1
11
12 result = firstUniqChar(s)
13 print(result)
```

-1

In []:

1