242. Valid Anagram

Add to List ▼

Question Editorial Solution

My Submissions (/problems/valid-anagram/submissions/)

Total Accepted: 132753 Total Submissions: 295290 Difficulty: Easy Contributors: Admin

Given two strings *s* and *t*, write a function to determine if *t* is an anagram of *s*.

For example,

```
s = "anagram", t = "nagaram", return true.s = "rat", t = "car", return false.
```

Note:

You may assume the string contains only lowercase alphabets.

Follow up:

What if the inputs contain unicode characters? How would you adapt your solution to such case?

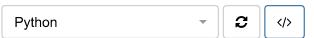
Subscribe (/subscribe/) to see which companies asked this question

Show Similar Problems

Have you met this question in a real interview? Yes No

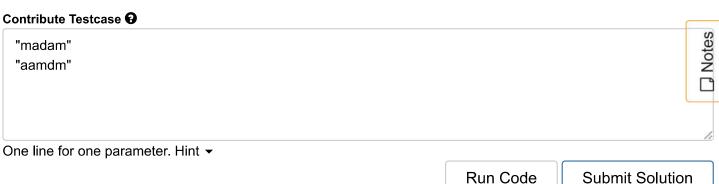
Discuss (https://discuss.leetcode.com/category/303) Top Solutions

Pick One (/problems/random-one-question/)

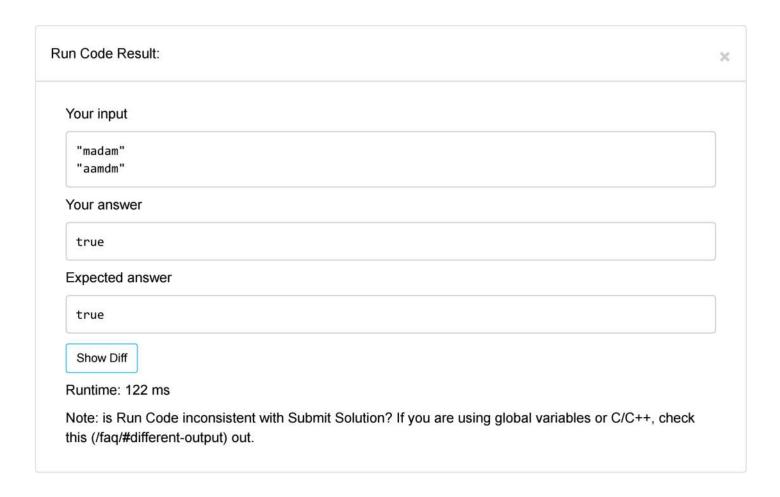


```
class Solution(object):
2
        def isAnagram(self, s, t):
3
4
             :type s: str
5
             :type t: str
6
             :rtype: bool
7
            letters=[0]*26
8
9
10
            while i<len(s):
11
                 index=ord(s[i])-97
12
                 letters[index]=letters[index]+1
13
                 i=i+1
            i=0
14
15
            while i<len(t):
                 index=ord(t[i])-97
16
```

Custom Testcase



Run Code Status: Finished



Frequently Asked Questions (/faq/) | Terms of Service (/tos/)