Unique Characters in String

Problem

Given a string,determine if it is compreised of all unique characters. For example, the string 'abcde' has all unique characters and should return True. The string 'aabcde' contains duplicate characters and should return false.

Solution

We'll show two possible solutions, one using a built-in data structure and a built in function, and another using a built-in data structure but using a look-up method to check if the characters are unique.

```
In [1]:  def uni_char(s):
             return len(set(s)) == len(s)
```

```
In [2]:  def uni_char2(s):
             chars = set()
             for let in s:
                 # Check if in set
                 if let in chars:
                     return False
                 else:
                     #Add it to the set
                     chars.add(let)
             return True
```

Test Your Solution

```
In [3]:  """
         RUN THIS CELL TO TEST YOUR CODE>
         """
         from nose.tools import assert_equal


         class TestUnique(object):

             def test(self, sol):
                 assert_equal(sol(''), True)
                 assert_equal(sol('goo'), False)
                 assert_equal(sol('abcdefg'), True)
                 print('ALL TEST CASES PASSED')

         # Run Tests
         t = TestUnique()
         t.test(uni_char)
```

```
ALL TEST CASES PASSED
```

Good Job!