AlgoExpert Quad Layout Python 14px Sublime 00:00:00

Our Solution(s) Run Code

Solution 1

```python
# Copyright © 2020 AlgoExpert, LLC. All rights reserved.

# O(n) time | O(n) space
def balancedBrackets(string):
    openingBrackets = "([{"
    closingBrackets = ")]}"
    matchingBrackets = {")": "(", "]": "[", "}": "{"}
    stack = []
    for char in string:
        if char in openingBrackets:
            stack.append(char)
        elif char in closingBrackets:
            if len(stack) == 0:
                return False
            if stack[-1] == matchingBrackets[char]:
                stack.pop()
            else:
                return False
    return len(stack) == 0

```

Your Solutions Run Code

Solution 1 Solution 2 Solution 3

```python
def balancedBrackets(string):
    # Write your code here.
    pass

```

Our Tests

```python
import program
```

Custom Output Submit Code

```
import unittest


class TestProgram(unittest.TestCase):
    def test_case_[illegible](self):
        self.assertEqual(program.balancedBrackets("[illegible]"), Tr

    def test_case_[illegible](self):
        self.assertEqual(program.balancedBrackets("[illegible]

    def test_case_[illegible](self):
        self.assertEqual(program.balancedBrackets("[illegible]")

    def test_case_[illegible](self):
        self.assertEqual(program.balancedBrackets("[illegible]

    def test_case_[illegible](self):
        self.assertEqual(program.balancedBrackets("[illegible]"), True

    def test_case_[illegible](self):
        self.assertEqual(program.balancedBrackets("[illegible]"), Tr
```

Run or submit code when you're ready.