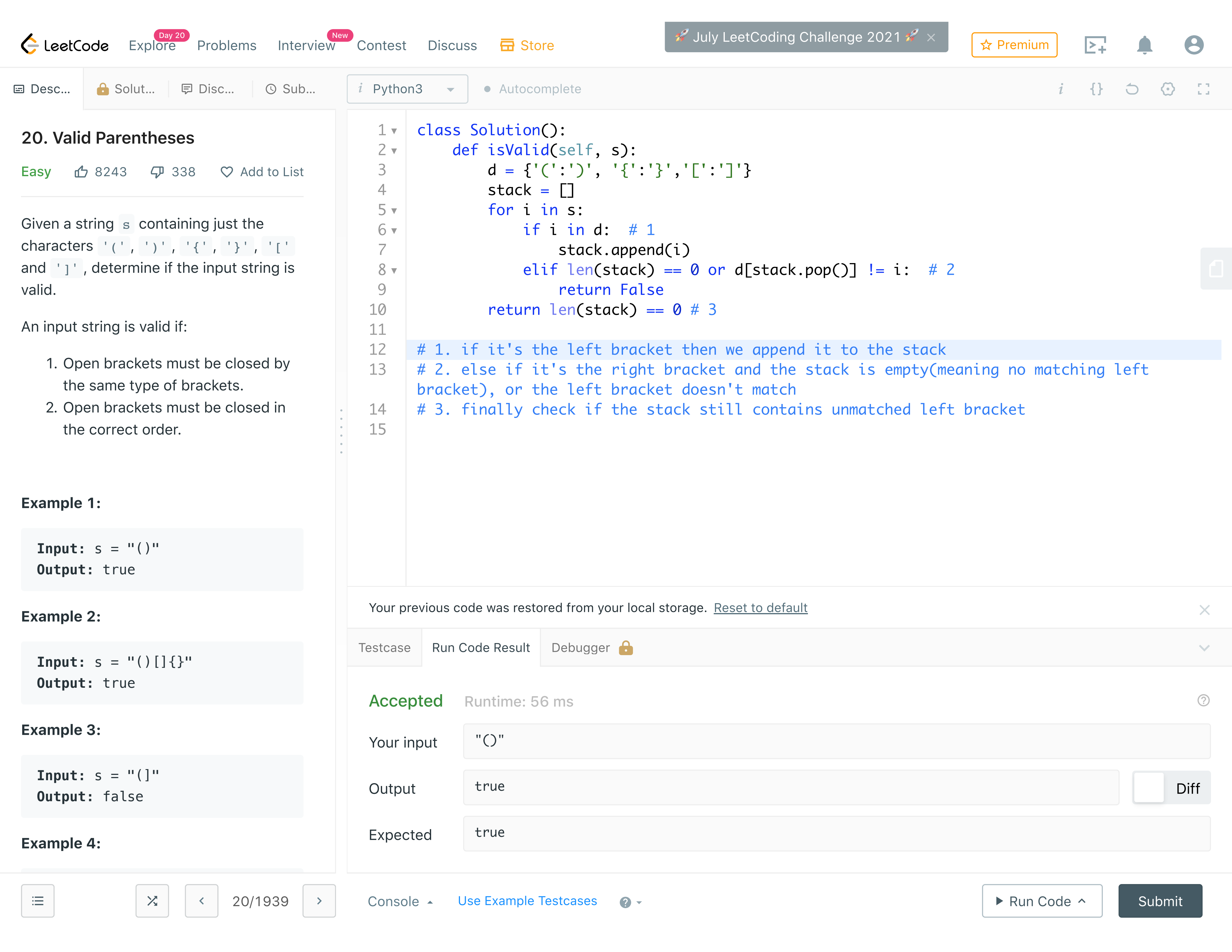

20. Valid Parentheses

Easy 8243 338 Add to List

Given a string `s` containing just the characters `'('`, `')'`, `'{'`, `'}'`, `'['` and `']'`, determine if the input string is valid.

An input string is valid if:

1. Open brackets must be closed by the same type of brackets.
2. Open brackets must be closed in the correct order.

Example 1:

```
Input: s = "()"
Output: true
```

Example 2:

```
Input: s = "()[]{}"
Output: true
```

Example 3:

```
Input: s = "(]"
Output: false
```

Example 4:

```
class Solution():
    def isValid(self, s):
        d = {'(':')', '{':'}','[':']'}
        stack = []
        for i in s:
            if i in d:  # 1
                stack.append(i)
            elif len(stack) == 0 or d[stack.pop()] != i:  # 2
                return False
        return len(stack) == 0 # 3

# 1. if it's the left bracket then we append it to the stack
# 2. else if it's the right bracket and the stack is empty(meaning no matching left bracket), or the left bracket doesn't match
# 3. finally check if the stack still contains unmatched left bracket
```

Your previous code was restored from your local storage. Reset to default

Accepted Runtime: 56 ms

Your input `"()"`

Output `true` Diff

Expected `true`