

0013 - Roman to Integer








Symbol	Value
I	1
V	5
X	10
L	50
C	100
D	500
M	1000



- **I** can be placed before **V** (5) and **X** (10) to make 4 and 9.
- **X** can be placed before **L** (50) and **C** (100) to make 40 and 90.
- **C** can be placed before **D** (500) and **M** (1000) to make 400 and 900.



- 
 - 


 - 
- total



```
class Solution(object):
    def romanToInt(self, s):
        """
        :type s: str
        :rtype: int
        """
        total = 0
```

```

formar_char = ''

for char in s:
    if char == 'I':
        total += 1
    elif char == 'V':
        if formar_char == 'I':
            total += 3 # [] 4 [] [] [] [] [] [] [] [] I [] [] [] [] XD [] [] [] [] [] [] [] []
        else:
            total += 5
    elif char == 'X':
        if formar_char == 'I':
            total += 8
        else:
            total += 10
    elif char == 'L':
        if formar_char == 'X':
            total += 30
        else:
            total += 50
    elif char == 'C':
        if formar_char == 'X':
            total += 80
        else:
            total += 100
    elif char == 'D':
        if formar_char == 'C':
            total += 300
        else:
            total += 500
    elif char == 'M':
        if formar_char == 'C':
            total += 800
        else:
            total += 1000
    formar_char = char
return total

```



if-else yandev...XD
Map Map iterator for

```
class Solution(object):
    def romanToInt(self, s):
        """
        :type s: str
        :rtype: int
        """
        total = 0

        roman_map = {
            'I': 1,
            'V': 5,
            'X': 10,
            'L': 50,
            'C': 100,
            'D': 500,
            'M': 1000
        }

        for i in range(0, len(s)):
            if i != 0 and roman_map[s[i - 1]] < roman_map[s[i]]:
                # 
                total = total + roman_map[s[i]] - (roman_map[s[i - 1]] * 2)
            else:
                total += roman_map[s[i]]

        return total
```

str.replace() ...

Java

```
class Solution {
    public int romanToInt(String s) {
        HashMap<Character, Integer> romanMap = new HashMap<>();
```

```
romanMap.put('I', 1);
romanMap.put('V', 5);
romanMap.put('X', 10);
romanMap.put('L', 50);
romanMap.put('C', 100);
romanMap.put('D', 500);
romanMap.put('M', 1000);

int total = 0;
for(int count = 0; count < s.length(); count++){
    if(count != 0 && romanMap.get(s.charAt(count)) > romanMap.get(s.charAt(count - 1))){
        total -= romanMap.get(s.charAt(count - 1)) * 2;
    }

    total += romanMap.get(s.charAt(count));
}

return total;
}
}
```

Revision #3

Created 2 July 2024 05:42:48 by Nesquate

Updated 8 July 2024 03:17:01 by Nesquate