

TASK – 5

Aim :

Complete the merge_the_tools function in the editor below.

merge_the_tools has the following parameters:

string s: the string to analyze

int k: the size of substrings to analyze

Prints :

Print each subsequence on a new line. There will be of them. No return value is expected. Returns :

string: a single string made up of each of the lines of the rangoli separated by a newline character (\n)

Code Implementation :

```
def merge_the_tools(s, k):
    for i in range(0, len(s), k):
        substring = s[i:i+k]
        seen = ''
        for char in substring:
            if char not in seen:
                seen += char
        print(seen)

if __name__ == '__main__':
    string, k = input(), int(input())
    merge_the_tools(string, k)
```

Output :

The screenshot shows a code editor interface with a sidebar on the left containing a list of test cases, all marked as successful with green checkmarks. The main area displays the compiler's output, which includes a 'Success' message, the input string 'AABCAAADA' and integer '3', and the expected output consisting of three lines: 'AB', 'CA', and 'AD'.

Test case 0	
✓	Test case 0
✓	Test case 1
✓	Test case 2
✓	Test case 3
✓	Test case 4
✓	Test case 5
✓	Test case 6

Compiler Message

Success

Input (stdin) [Download](#)

1	AABCAAADA
2	3

Expected Output [Download](#)

1	AB
2	CA
3	AD

