

Camel Case is a naming style common in many programming languages. In Java, method and variable names typically start with a lowercase letter, with all subsequent words starting with a capital letter (example: startThread). Names of classes follow the same pattern, except that they start with a capital letter (example: BlueCar).

Your task is to write a program that creates or splits Camel Case variable, method, and class names.

Input Format

- Each line of the input file will begin with an operation (S or C) followed by a semi-colon followed by M, C, or V followed by a semi-colon followed by the words you'll need to operate on.

- The operation will either be S (split) or C (combine)

- M indicates method, C indicates class, and V indicates variable

- In the case of a split operation, the words will be a camel case method, class or variable name that you need to split into a space-delimited list of words starting with a lowercase letter.

- In the case of a combine operation, the words will be a space-delimited list of words starting with lowercase letters that you need to combine into the appropriate camel case String. Methods should end with an empty set of parentheses to differentiate them from variable names.

Output Format

- For each input line, your program should print either the space-delimited list of words (in the case of a split operation) or the appropriate camel case string (in the case of a combine operation).

Sample Input

S;M;plasticCup()

C;V;mobile phone

C;C;coffee machine

S;C;LargeSoftwareBook

C;M;white sheet of paper

S;V;pictureFrame

Sample Output

plastic cup

mobilePhone

CoffeeMachine

large software book

whiteSheetOfPaper()

picture frame

Explanation

- Use Scanner to read in all information as if it were coming from the keyboard.

- Print all information to the console using standard output (System.out.print() or System.out.println()).

- Outputs must be exact (exact spaces and casing).

```
1 inputs = []
2
3 while True:
4     try:
5         instruction = input()
6         inputs.append(instruction)
7     except:
8         break
9
10 for idx, instruction in enumerate(inputs):
11     # necessary step other-wise there are errors in parsing the input
12     inputs[idx] = inputs[idx].strip('\n\r')
13
14 def split_function(string, operand_type='V'):
15     capitals = [0]
16     for idx in range(1, len(string)):
17         if string[idx].isupper():
18             capitals.append(idx)
19
20     result = ''
21     for parts_idx in range(1, len(capitals)):
22         result += string[capitals[parts_idx-1]:capitals[parts_idx]] + ' '
23
24     result += string[capitals.pop():]
25
26 if operand_type == 'M':
27     result = result[:-2]
28
29 return result.lower()
30
31 def combine(string, taip='V'):
32     result = ''
33     words = string.split()
34     for idx in range(len(words)):
35         words[idx] = words[idx][0].upper() + words[idx][1:]
36         result += words[idx]
37
38 if taip == 'M':
39     result = result[0].lower() + result[1:] + '()'
40 elif taip == 'V':
41     result = result[0].lower() + result[1:]
42 else:
43     pass
44
45 return result
46
47 if __name__ == '__main__':
48     for instruction in inputs:
49         operation = instruction[0]
50         operand_type = instruction[2]
51         operand = instruction[4:]
52
53 if operation == 'S':
54     result = split_function(operand, operand_type)
55
56 if operation == 'C':
57     result = combine(operand, operand_type)
58
59 print(result)
60
61
62
```

Line: 62 Col: 1

 Upload Code as File ☐ Test against custom input

Run Code

Submit Code

Congratulations

You solved this challenge. Would you like to challenge your friends?



Next Challenge

Test case 0

Test case 1

Compiler Message

Success

Input (stdin)

Download

```
1 S;V;iPad
2 C;M;mouse pad
3 C;C;code swarm
4 S;C;OrangeHighlighter
```

Expected Output

Download

```
plastic cup
mobilePhone
CoffeeMachine
large software book
whiteSheetOfPaper()
picture frame
```