In [ ]:

```
# Strings
```

In [1]:

```
# Swap Case

```

In [ ]:

```
def swap_case(s):
    num = ""
    for let in s:
        if let.isupper() == True:
            num+=(let.lower())
        else:
            num+=(let.upper())
    return num



if __name__ == '__main__':
    s = input()
    result = swap_case(s)
    print(result)
```

In [2]:

```
# String Split and Join
```

In [ ]:

```
def split_and_join(line):
    Output = line.split();
    Output = "-".join(Output)
    return Output;

if __name__ == '__main__':
    line = input()
    result = split_and_join(line)
    print(result)
```

In [3]:

```
# Whats your name
```

In [ ]:

```
def print_full_name(a, b):
    print("Hello %s %s! You just delved into python."%(a,b))


if __name__ == '__main__':
    first_name = input()
    last_name = input()
    print_full_name(first_name, last_name)
```

In [4]:

```
# Mutations
```

In [ ]:

```
def mutate_string(string, position, character):
    lis=list(string)
    lis[position]=character
    return ''.join(lis)

if __name__ == '__main__':
    s = input()
    i, c = input().split()
    s_new = mutate_string(s, int(i), c)
    print(s_new)
```

In [ ]:

```
# Find a String
```

In [ ]:

```
def count_substring(string, sub_string):
    count=0
    for i in range(len(string)):
        for j in range(len(sub_string)):
            if string[i+j]==sub_string[j] and j==(len(sub_string)-1):
                count=count+1
            if string[i+j]!=sub_string[j]:
                break
        if i==len(string)-len(sub_string):
            break
    return count
if __name__ == '__main__':
    string = input().strip()
    sub_string = input().strip()

    count = count_substring(string, sub_string)
    print(count)
```

In [5]:

```
# String Validators
```

In [ ]:

```
if __name__ == '__main__':
    s = input()
    print(any(map(str.isalnum, s)))
    print(any(map(str.isalpha, s)))
    print(any(map(str.isdigit, s)))
    print(any(map(str.islower, s)))
    print(any(map(str.isupper, s)))
```

In [ ]:

```
# Text Alignment
```

In [ ]:

```
thickness = int(input()) #This must be an odd number
c = 'H'
#Top Cone
for i in range(thickness):
    print((c*i).rjust(thickness-1)+c+(c*i).ljust(thickness-1))
#Top Pillars
for i in range(thickness+1):
    print((c*thickness).center(thickness*2)+(c*thickness).center(thickness*6))
#Middle Belt
for i in range((thickness+1)//2):
    print((c*thickness*5).center(thickness*6))
#Bottom Pillars
for i in range(thickness+1):
    print((c*thickness).center(thickness*2)+(c*thickness).center(thickness*6))
#Bottom Cone
for i in range(thickness):
    print(((c*(thickness-i-1)).rjust(thickness)+c+(c*(thickness-i-1)).ljust(thickness)).r
```

In [6]:

```
# Design door mat
```

In [ ]:

```
n, m = map(int,input().split())
pattern = [('.|.'*(2*i + 1)).center(m, '-') for i in range(n//2)]
print('\n'.join(pattern + ['WELCOME'.center(m, '-')] + pattern[::-1]))
```

In [7]:

```
# String Formatting
```

In [ ]:

```
def print_formatted(number):

    l1 = len(bin(number)[2:])

    for i in range(1,number+1):
        print(str(i).rjust(l1,' '),end=" ")
        print(oct(i)[2:].rjust(l1,' '),end=" ")
        print(((hex(i)[2:]).upper()).rjust(l1,' '),end=" ")
        print(bin(i)[2:].rjust(l1,' '),end=" ")
        print("")
if __name__ == '__main__':
    n = int(input())
    print_formatted(n)
```

In [8]:

```
# Alphabet rangoli
```

In [ ]:

```
def print_rangoli(size):

    import string
    design = string.ascii_lowercase
    L = []
    for i in range(n):
        s = "-".join(design[i:n])
        L.append((s[::-1]+s[1:]).center(4*n-3, "-"))

    print('\n'.join(L[:0:-1]+L))

if __name__ == '__main__':
    n = int(input())
    print_rangoli(n)
```

In [9]:

```
# Capitalize
```

In [ ]:

```
def solve(s):
    for x in s[:].split():
        s = s.replace(x, x.capitalize())
    return s

if __name__ == '__main__':
    fptr = open(os.environ['OUTPUT_PATH'], 'w')

    s = input()

    result = solve(s)

    fptr.write(result + '\n')

    fptr.close()
```

In [10]:

```
# The minion game
```

In [ ]:

```
def minion_game(string):

    player1 = 0;
    player2 = 0;
    str_len = len(string)
    for i in range(str_len):
        if s[i] in "AEIOU":
            player1 += (str_len)-i
        else :
            player2 += (str_len)-i

    if player1 > player2:
        print("Kevin", player1)
    elif player1 < player2:
        print("Stuart",player2)
    elif player1 == player2:
        print("Draw")
    else :
        print("Draw")

if __name__ == '__main__':
    s = input()
    minion_game(s)
```

In [11]:

```
# Merge the tools
```

In [ ]:

```
def merge_the_tools(string, k):

    temp = []
    len_temp = 0
    for item in string:
        len_temp += 1
        if item not in temp:
            temp.append(item)
        if len_temp == k:
            print (''.join(temp))
            temp = []
            len_temp = 0
if __name__ == '__main__':
    string, k = input(), int(input())
    merge_the_tools(string, k)
```