itertools.combinations() ★

Your itertools.combinations() submission got 10.00 points.   Share   Post

Try the next challenge | **Try a Random Challenge**

Problem   Submissions   Leaderboard   Editorial 🔒

itertools.combinations(iterable, r)

This tool returns the r length subsequences of elements from the input iterable.

Combinations are emitted in lexicographic sorted order. So, if the input iterable is sorted, the combination tuples will be produced in sorted order.

Sample Code

```
>>> from itertools import combinations
>>>
>>> print list(combinations('12345',2))
[('1', '2'), ('1', '3'), ('1', '4'), ('1', '5'), ('2', '3'), ('2', '4'), ('2', '5'), ('3', '4'), ('3', '5'), ('4', '5')]
>>>
>>> A = [1,1,3,3,3]
>>> print list(combinations(A,4))
[(1, 1, 3, 3), (1, 1, 3, 3), (1, 1, 3, 3), (1, 3, 3, 3), (1, 3, 3, 3)]
```

Task

You are given a string S.

Your task is to print all possible combinations, up to size k, of the string in lexicographic sorted order.

Input Format

A single line containing the string S and integer value k separated by a space.

Constraints

$0 < k \leq len(S)$

The string contains only UPPERCASE characters.

Output Format

Print the different combinations of string S on separate lines.

Sample Input

```
HACK 2
```

Sample Output

```
A
C
H
K
AC
AH
AK
CH
```

CK
HK

Change Theme   Language   Python 3

```python
# Enter your code here. Read input from STDIN. Print output to STDOUT
from itertools import combinations

string, k = input().split()
string = sorted(string)
k = int(k)

for i in range(1, k+1):
    lst = list(combinations(string, i))

    for j in lst:
        print("".join(j))


```

EMACS   Line: 14 Col: 1

⬆ Upload Code as File   ☐ Test against custom input   Run Code   Submit Code

You have earned 10.00 points!
50/115 challenges solved.

43%

Congratulations

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

<div style="text-align: right;">
Next Challenge
</div>

⊘	**Test case 0**		Compiler Message
⊘	Test case 1	🔒	Success
⊘	Test case 2	🔒	Input (stdin) Download
⊘	Test case 3	🔒	1 HACK 2
⊘	Test case 4	🔒	Expected Output Download
⊘	Test case 5	🔒	1 A

Expected Output:

```
1    A
2    C
3    H
4    K
5    AC
6    AH
```