# Notes :

from itertools import combinations  # Importing combinations function from itertools module

input\_string = input()

# Split the input string into separate inputs

S, n = input\_string.split()

sorted\_string = ''.join(sorted(S))

for i in range((len(sorted\_string))):

    print(sorted\_string[i], end='\n')

# Generate combinations and print each combination

for combination in combinations(sorted\_string, int(n)):

    print(''.join(combination))  # Joining the elements of each combination tuple and printing it

Code modifie pour etre plus inclusive :

from itertools import combinations  # Importing combinations function from itertools module

input\_string = input()

# Split the input string into separate inputs

S, n = input\_string.split()

sorted\_string = ''.join(sorted(S))

for i in range((len(sorted\_string))):

    print(sorted\_string[i], end='\n')

# Generate combinations and print each combination

for size in range(2, int(n) + 1):

    for combination in combinations(sorted\_string, size):

        print(''.join(combination))  # Joining the elements of each combination tuple and printing it

1. itertools.combinations\_with\_replacement()

from itertools import \*  # Importing combinations function from itertools module

input\_string = input()

# Split the input string into separate inputs

S, n = input\_string.split()

sorted\_string = ''.join(sorted(S))

# Generate combinations and print each combination

for size in range(int(n), int(n) + 1):

    for combination in combinations\_with\_replacement(sorted\_string, size):

        print(''.join(combination))  # Joining the elements of each combination tuple and printing it

[Itertool with groupby()](https://docs.python.org/2/library/itertools.html#itertools.groupby)

***Even odd using FOR loop :***

#include <stdio.h>

void numberstr(int n){

    switch(n) {

         case 1:

            printf("one\n");

            break;

        case 2:

            printf("two\n");

            break;

        case 3:

            printf("three\n");

            break;

        case 4:

            printf("four\n");

            break;

        case 5:

            printf("five\n");

            break;

        case 6:

            printf("six\n");

            break;

        case 7:

            printf("seven\n");

            break;

        case 8:

            printf("eight\n");

            break;

        case 9:

            printf("nine\n");

            break;

    }

}

int main()

{

    int a, b;

    scanf("%d\n%d", &a, &b);

    for(int n = a; n<=b; n++){

        if (n>=1 && n<=9){

            numberstr(n);

        }

        else if (n>9)

        {

            if (n%2==0){

                printf("even\n");

            }

            else if (n%2!=0)

            {

                printf("odd\n");

            }

        }

    }

}

# !!!To have input on a single line :

n=int(input())

c=input()

runnerUp = list(map(int, c.split()))

 **user\_input.split()**:

* This splits the input string user\_input into a list of substrings based on spaces.
* For example, if user\_input is "10 20 30 20 40", user\_input.split() returns ['10', '20', '30', '20', '40'].

 **map(int, user\_input.split())**:

* map(int, user\_input.split()) applies the int function to each item in the list ['10', '20', '30', '20', '40'].
* This converts each string in the list to an integer.
* The result is a map object where each item is an integer.

 **list(map(int, user\_input.split()))**:

* The map object is then converted to a list using list().
* The final result is a list of integers: [10, 20, 30, 20, 40]