



Python

Урок 3

Функции

```
print(input("Enter name >> ").title())
```

```

1  from math import factorial
2
3  n = int(input("Enter number n >> "))
4  print("(a + b)^%d = " % n, end='')
5
6  for i in range(n + 1):
7      s = ""
8
9      c = int(factorial(n) / factorial(n - i) / factorial(i))
10     if(c != 1):
11         s += str(c) + "*"
12
13     if(i != n):
14         s += "a"
15         s += "^"+str(n - i) if i != n - 1 else ""
16         s += "*" if i != 0 else ""
17
18     if(i != 0):
19         s += "b"
20         s += "^"+str(i) if i != 1 else ""
21
22     if(i != n):
23         print(s + " + ", end='')
24     else:
25         print(s)

```

$$(a + b)^n = \sum_{k=0}^n C_n^k a^{n-k} b^k$$

$$C_m^k = \frac{m!}{k!(m-k)!}$$

```
1 import string
2 s = input("Enter text lower\n\n"+" " * 7 + "||\n" + " " * 7 + "\\\n\n")
3 formatted = s.replace(".", "")
4
5 print("\nThe length of this extract is",
6       | len(formatted.replace(",","")) - len(formatted.split()) + 1, "letters")
7 print("There are", len(s.split()), "words and", s.count("."), "sentences")
8
9 formatted = formatted.lower()
10 print("The lowercase text is:\n" + formatted)
11
12 letters = {}
13
14 for i in "abcdefghijklmnopqrstuvwxyz":
15     | letters[i] = 0
16
17 for i in formatted:
18     | if i != " ":
19     |     | letters[i] += 1
20
21 for i in letters.keys():
22     | if letters[i] != 0:
23     |     | print("The number of letters \'%s\' is %d" % (i, letters[i]))
```

- Тернарные операторы

`condition_is_true if condition else condition_is_false`

■ Синтаксис

```
def function(arg1, arg2,...):
```

```
...
```

```
...
```

■ Ваши аргументы?

```
def function():
```

```
...
```

```
...
```

■ Значение

```
def cube(x):  
    return x ** 3;
```

```
def hello(a):  
    print("Hello, ", a)
```


■ Задание

- Напишите функцию поиска максимума в списке

■ Программа

```
def max(a):  
    max = a[0]  
    for i in a[1:]:  
        if(i > max):  
            max = i  
    return max
```

■ Больше аргументов

```
def display_message(message, truncate_after=4):  
    print message[:truncate_after]
```

```
>> display_message("message")
```

```
"mess"
```

```
>> display_message("message", 6)
```

```
"messag"
```

■ Больше аргументов

```
def fun(*la):  
    print(la)
```

```
fun(1, 2)
```

```
 #(1, 2)
```

■ Больше звёздочек!

```
def fun(**la):  
    for i in la.keys():  
        print(i, " - ", la[i])
```

```
fun(key1 = 2, key2 = 2)
```

```
#key1 - 2
```

```
#key2 - 2
```

- И наконец...

```
def a(a,b):  
    print(a,b)  
print(a(b = 1, a = 2))  
  
#“2 1”
```

■ Задание

- Напишите аналог `print` (но назовите по-другому), выводящий каждый аргумент с пометкой, какой это аргумент (его номер)

■ Программа

```
def printt(*la):  
    for i in range(len(la)):  
        print("Argument number %d - %d" % (i+1,la[i]))
```


■ Лямбда выражения

```
add = lambda x, y: x + y
```

```
print(add(2,3))
```

```
#5
```

■ Задача

- Написать функцию, вычисляющую сумму произведения двух чисел с числом, возведённым в степень, равную второму числу.

- Программа

```
superFunction = lambda a, b: a ** b + a * b
```