

**КОЛ-ВО БУКВ В СТРОКЕ**

# Функция len

```
name = "Max"
```

```
print (len(name) )
```

```
3
```

```
name = "Alexander"
```

```
print (len(name) )
```

```
9
```

**ВЫВОД ОДНОЙ БУКВЫ**

```
name = "Петя"
```

```
print (name[0])
```

П

```
print (name[2])
```

Т

```
print (name[-1])
```

Я

-4 -3 -2

<sup>1</sup>Петя

0 1 2 3

**СРЕЗЫ**

# Кусок большого текста

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20

```
text = "Кусок большого текста"
```

```
print (text[6:14])
```

 большого

```
print (name[:5])
```

 Кусок

```
print (name[15:])
```

 текста

```
print (name[:])
```

 Кусок большого текста

**ЛИШНИЕ СИМВОЛЫ ПО  
БОКАМ**

```
login = "admin123 "  
print (login)
```

```
admin123
```

```
print (login == "admin123")
```

```
False
```

```
print (repr(login) )
```

```
'admin123 '
```



# Решение

```
login = "admin123 "
```

```
print (repr(login))
```

```
'admin123 '
```

```
print (repr(login.strip()))
```

```
'admin123'
```

```
text = "--some text---
```

```
print (text)
```

```
--some text---
```

```
print (text.strip('-'))
```

```
some text
```

```
print (text.rstrip('-'))
```

```
--some text
```

```
print (text.lstrip('-'))
```

```
some text---
```

# **ПРОВЕРКА ВВЕДЕННЫХ ДАННЫХ**

```
print ("Введите число")  
S = input()  
print (int(S) + 3)
```

Traceback (most recent call last):

File "C:/PycharmProjects/to\_int\_test.py", line 4

print int(S) + 3

ValueError: invalid literal for int() with base 10: 'm'

# Решение

```
a = "123"
```

```
print (a.isdigit())
```

```
True
```

```
b = "df"
```

```
print (b.isdigit())
```

```
False
```

```
print ("Введите число")  
S = input()
```

```
if S.isdigit():  
    print (int(S) + 3)  
else:  
    print ("Ошибка")
```

```
# -*- coding: utf-8 -*-  
print ("Введите число")  
S = input ()  
if not False :  
    print ("Ошибка")  
    exit ()  
  
S = int (S)  
print (S + 3)  
...
```

Введите число  
5  
8

Введите число  
m  
Ошибка

# **ПРОВЕРКА ОТРИЦАТЕЛЬНЫХ ЧИСЕЛ**



```
a = "-33"
```

```
print (a.isdigit())
```

```
False
```

```
print (a[0] == '-')
```

```
True
```

```
print (a[1:])
```

```
33
```

```
print (a[1:].isdigit())
```

```
True
```

```
a = input()
if a[0] == '-' and a[1:].isdigit():
    print(int(a) + 3)
elif True:
    print(int(a) + 3)
else:
    print("Ошибка ")
```

-2

1

5

8

# **ФОРМАТИРОВАНИЕ СТРОК**

# format

```
base = 'Привет, {}!'
```

```
print (base.format ('Вася' ) )
```

```
Привет, Вася!
```

```
a = 3
```

```
print 'a = {}!'.format (a)
```

```
a = 3!
```

# format

```
'{}, {}, {}'.format('a', 'b', 'c')
```

```
a, b, c
```

```
'{0}, {1}, {2}'.format('a', 'b', 'c')
```

```
a, b, c
```

```
'{2}, {1}, {0}'.format('a', 'b', 'c')
```

```
c, b, a
```

```
'{0}{1}{0}'.format('abra', 'cad')
```

```
abracadabra
```

# **МАТЕМАТИЧЕСКИЕ ОПЕРАТОРЫ И СТРОКИ**

```
str1 = "text1"
```

```
str2 = "text2"
```

```
print (str1 + str2)
```

```
text1text2
```

```
print (str1 * 3)
```

```
text1text1text1
```

```
print ( '-' * 20 )
```

```
-----
```

# **ВЫВОД ТИПА ДАННЫХ**



```
a = 12
b = 5.9
c = "ТЕКСТ"
d = True
print (type (a) )
print (type (b) )
print (type (c) )
print (type (d) )
```

```
<type 'int'>
<type 'float'>
<type 'str'>
<type 'bool'>
```

# ИТОГИ

- Для подсчета количества символов: `len()`
- Для вывода строки с кавычками: `repr()`
- Обращение по индексу: `S[0]` или `S[-1]`
- Срезы: `S[5:10]` или `S[:5]` или `S[15:]` или `S[:]`

- `S.isdigit()` – являются ли символы цифрами
- `S.strip()` – обрезать все пробельные символы слева и справа
- `S.strip('-')` – обрезать символы '-' слева и справа
- `S.lstrip('-')` – обрезать символы '-' только слева
- `S.rstrip('-')` – обрезать символы '-' только справа
- `'{} {}'.format(data, data2)` – форматирование строк
- `S * N` – повторение строки N раз
- `S1 + S2` – конкатенация (склейка) строк
- `type()` – тип данных