



# Цикл For. Range. Работа со строками в Python.

# Цикл `for`

Этот цикл пробегается по набору значений, помещает каждое значение в переменную, и затем в цикле мы можем с этой переменной производить различные действия.

# Сравнение циклов

while

```
letters = ["А", "Б", "В", "Г"]  
  
i = 0  
while i < len(letters):  
    let = letters[i]  
    print(let)  
    i += 1  
  
print("Готово")
```

for

```
letters = ["А", "Б", "В", "Г"]  
  
for let in letters:  
    print(let)  
  
print("Готово")
```

# Сравнение циклов

while

```
letters = ["А", "Б", "В", "Г"]

i = 0
while i < len(letters):
    let = letters[i]
    print(let)
    i += 1

print("Готово")
```

for

```
letters = ["А", "Б", "В", "Г"]

for let in letters:
    print(let)

print("Готово")
```

# Сравнение циклов

while

```
letters = ["А", "Б", "В", "Г"]

i = 0
while i < len(letters):
    let = letters[i]
    print(let)
    i += 1

print("Готово")
```

for

```
letters = ["А", "Б", "В", "Г"]

for let in letters:
    print(let)

print("Готово")
```

# Сравнение циклов

while

```
letters = ["А", "Б", "В", "Г"]  
  
i = 0  
while i < len(letters):  
    let = letters[i]  
    print(let)  
    i += 1  
  
print("Готово")
```

for

```
letters = ["А", "Б", "В", "Г"]  
  
for let in letters:  
    print(let)  
  
print("Готово")
```

# Перебор списка или строки

```
string = "АБВГ"  
  
for st in string:  
    print(f"Буква: {st}")
```

```
# Буква: А  
# Буква: Б  
# Буква: В  
# Буква: Г
```

```
arr = [9, 4, 5, 2]  
  
for a in arr:  
    print(f"Цифра: {a}")
```

```
# Цифра: 9  
# Цифра: 4  
# Цифра: 5  
# Цифра: 2
```

# Повторение строк в print()

```
print("a")  
# a
```

```
print("a" * 3)  
# aaa
```

```
print("he" * 4)  
# hehehehe
```

```
print([1, 2, 3])  
# [1, 2, 3]
```

```
print([1, 2, 3] * 2)  
# [1, 2, 3, 1, 2, 3]
```

```
print(["Ф", True] * 2)  
# ['Ф', True, 'Ф', True]
```

# Диапазоны range()

Диапазоны или range представляют неизменяемый последовательный набор чисел.

range(stop) возвращает все целые числа от 0 до stop (не включая)

range(start, stop) возвращает все целые числа в промежутке от start (включая) до stop (не включая)

range(start, stop, step) возвращает целые числа в промежутке от start (включая) до stop (не включая), которые увеличиваются на значение step

# Диапазоны range()

```
range(5)           # 0, 1, 2, 3, 4  
range(1, 5)        # 1, 2, 3, 4  
range(2, 10, 2)    # 2, 4, 6, 8  
range(10, 2, -2)   # 10, 8, 6, 4
```

# Срезы строк

**Срезы для строк** — это механизм, с помощью которого извлекается подстрока по указанным параметрам.

```
string[начало:конец:шаг]
```

# Примеры срезов строк

```
string = "Привет мир!"  
  
# с 0 до 6 индекса - "Привет "  
print(string[:6])  
  
# с 7 до 10 индекса - "мир!"  
print(string[7:10])  
  
# с 0 индекса до конца - "Привет!"  
print(string[0::2])
```

```
arr = [1, 2, 3, 4]  
  
# с 1 до -1 индекса - [2, 3]  
print(arr[1:-1])  
  
# [4, 3, 2, 1]  
print(arr[::-1])
```

# Поиск в строке или списке

```
string = "Привет мир!"
```

```
exist1 = "мир" in string  
print(exist1) # True
```

```
exist2 = "ёж" in string  
print(exist2) # False
```

```
arr = [99, 22, 55, "444"]
```

```
exist1 = 99 in arr  
print(exist1) # True
```

```
exist2 = 444 in arr  
print(exist2) # False
```

# Методы работы со строками

<code>str.isalpha()</code>	Возвращает True, если строка состоит только из алфавитных символов
<code>str.islower()</code>	Возвращает True, если строка состоит только из символов в нижнем регистре
<code>str.isupper()</code>	Возвращает True, если все символы строки в верхнем регистре
<code>str.isdigit()</code>	Возвращает True, если все символы строки – цифры
<code>str.isnumeric()</code>	Возвращает True, если строка представляет собой число
<code>str.startswith(str2)</code>	Возвращает True, если строка начинается с подстроки <code>str2</code>
<code>str.endswith(str2)</code>	Возвращает True, если строка заканчивается на подстроку <code>str2</code>
<code>str.lower()</code>	Переводит строку в нижний регистр: “аБв” -> “абв”
<code>str.upper()</code>	Переводит строку в верхний регистр: “аБв” -> “АБВ”

# Методы работы со строками

<code>str.title()</code>	Начальные символы всех слов в строке переводятся в верхний регистр
<code>str.capitalize()</code>	Переводит в верхний регистр первую букву только первого слова строки
<code>str.lstrip()</code>	Удаляет начальные пробелы из строки: " абв " -> "абв "
<code>str.rstrip()</code>	Удаляет конечные пробелы из строки: "абв " -> "абв"
<code>str.strip()</code>	Удаляет все пробелы из строки: " абв " -> "абв"
<code>len(str)</code>	Считает количество символов в строке
<code>str.count(sub)</code>	Возвращает количество раз, которое подстрока <code>sub</code> встречается в строке <code>str</code>

# Методы работы со строками

<code>str.find(str2)</code>	Возвращает индекс подстроки <code>str2</code> в строке. Если подстрока не найдена, возвращает число <code>-1</code>
<code>str.join(strs)</code>	Объединяет строки в одну, вставляя между ними определённый разделитель
<code>str.split(delimiter)</code>	Разбивает строку на подстроку в зависимости от разделителя <code>delimiter</code> . Если разделителя нет, разбивает строку на символы
<code>str.replace(old, new)</code>	Заменяет в строке подстроку <code>old</code> на <code>new</code>

# Задания для выполнения

- 1) Создать список с любыми данными внутри и вывести все его элементы через цикл for
- 2) Создать строку с своей датой рождения и вывести число, месяц и год отдельно через срез
- 3) Создать строку "Hello, world!" и опробовать на нём любой метод из тех которые были изучены сегодня
- 4) Вывести число от 1 до 10 через цикл for при помощи range()

# Домашняя работа

- Написать программу для вычисления факториала на введённое от пользователя число
- Повторить пройденный материал