https://dfedorov.spb.ru

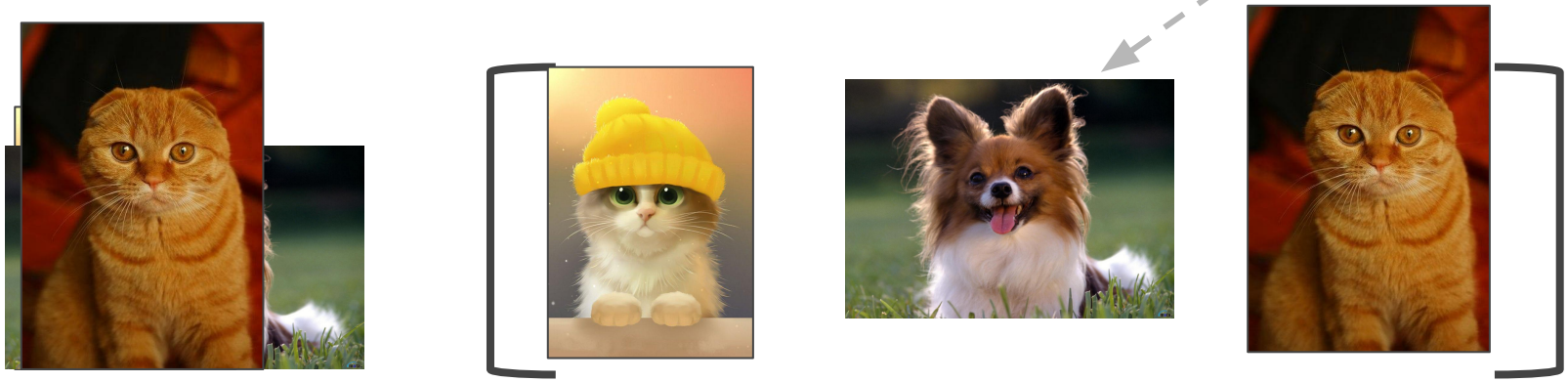
```
>>> for i in num:
        print(i, 'number')

0.8 number
7.0 number
6.8 number
-6 number
```

```
>>> num = [0.8, 7.0, 6.8, -6]
>>> for i in num:
        print(i, 'number')
```

Имя для переменной выбирает программист

Список num

i=



Общая форма для цикла for и списка

```
for переменная in список:
    тело цикла
```

4 пробела

Перебираем все объекты списка

Общая форма для цикла for и строки

```
for переменная in строка:
    тело цикла
```

Перебираем все символы строки

Найти в списке

i=

Сравниваем:

==

?

Пример

```
num = [0.8, 7.0, 6.8, -6]
for i in num:
    if i == 7.0:
        print(i, '- число 7.0')
```

```
country = "Russia"
for ch in country:
    if ch.isupper():
        print(ch)
```

посимвольно проверяем наличие буквы в верхнем регистре

0 1 2 3 4 5 6 7 8 9

начало

шаг 1

окончание

```
range(0, 10, 1)
```

не включительно

```
>>> for i in range(10):
        print(i, end=' ')
```

Constructor	Arguments	Result
`range(5)`	stop	0, 1, 2, 3, 4
`range(5, 10)`	start, stop	5, 6, 7, 8, 9
`range(10, 20, 2)`	start, stop, step	10, 12, 14, 16, 18

```
>>> range(1, 5)
range(1, 5)
>>> range(1, 5)[:2]
range(1, 3)

>>> list(range(1, 5))
[1, 2, 3, 4]
>>> len(range(1, 5))
4
>>> range(0, 3, 2) == range(0, 4, 2)
True
>>> range(-10, -2)
```

Документация

```python
N = 10 ** 12
for i in range(N):
    if i >= 10: break
    print(i, end=', ')
```

```
0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
```

Если бы диапазон фактически создавал этот список из одного триллиона значений, он занял бы десятки терабайт машинной памяти: пустая трата, учитывая тот факт, что мы игнорируем все, кроме первых 10 значений!
На самом деле, нет никакой причины, по которой итераторы вообще должны заканчиваться!

Библиотека Python **itertools** содержит функцию **count**, которая действует как бесконечный диапазон:

```
from itertools import count

for i in count():
    if i >= 10:
        break
    print(i, end=', ')
```

```
0, 1, 2, 3, 4, 5, 6, 7, 8, 9,
```

```
numbers = [3, 5, 9, -1, 3, 1]
result = 0
for item in numbers:
    if item < 0:
        break
    result += item
print(result)
```

выход из цикла

```
numbers = [3, 5, 9, -1, 3, 1]
result = 0
for item in numbers:
    if item < 0:
        continue
    result += item
print(result)
```

Подсчет суммы чисел на интервале от 1 до 100

```
>>> total = 0
>>> for i in range(1, 101):
            total = total + i  # total += i
>>> total
```

```
>>> sum(range(1, 101))
```

Python Way

https://dfedorov.spb.ru