

Python: Именованные аргументы

В этом уроке разберем, какие параметры существуют, чем они отличаются и в каких случаях их применять.

Аргументы — это данные, которые передаются в вызов функции. Они бывают двух типов:

Первый тип — **позиционные аргументы**. Они передаются в том же порядке, в котором определены параметры функции:

```
# (text, length)
truncate('My Text', 3)
```

Второй тип — **именованные аргументы**. Они передаются не просто как значения, а как пара «имя=значение». Поэтому их можно передавать в любом порядке:

```
# Аргументы переданы в другом порядке
truncate(length=3, text='My Text')
```

Если внимательно посмотреть на два примера выше, то можно понять, что это две одинаковые функции.

Теперь разберемся, в каких случаях нужно применять

эти типы аргументов.

Выбор типа параметра зависит от того, кто вызывает функцию.

Есть две причины использовать именованные аргументы:

- Они повышают читаемость, так как сразу видно имена
- Можно не указывать все промежуточные параметры, которые нам сейчас не нужны

Последнее полезно, если у функции много необязательных параметров. Посмотрим на примере:

```
def f(a=1, b=2, c=None, d=4):  
    print(a, b, c, d)
```

```
# Нужно передать только d, но приходится передавать все  
f(1, 2, 3, 8)
```

```
# Именованные аргументы позволяют передавать только d  
# Для остальных аргументов используются значения по умолчанию  
f(d=8)
```

Именованные аргументы можно передавать одновременно с позиционными. Тогда позиционные должны идти в самом начале:

```
# Передаем только a (позиционно) и d (как именованный)
f(3, d=3)
```

Задание

Реализуйте функцию `trim_and_repeat()`, которая принимает три параметра: строку, `offset` — число символов, на которое нужно обрезать строку слева и `repetitions` — количество раз, сколько ее нужно повторить, и возвращает получившуюся строку. Число символов для среза по умолчанию равно 0, а повторений — 1.

```
text = 'python'
print(trim_and_repeat(text, offset=3, repetitions=2)) # =>
print(trim_and_repeat(text, repetitions=3)) # => pythonpyth
print(trim_and_repeat(text)) # => python
```

► Упражнение не проходит проверку — что делать?



► В моей среде код работает, а здесь нет 🤔

► Мой код отличается от решения учителя 🤔

► Прочитал урок — ничего не понятно 🤔