

## Python X: map(), filter(), zip()

Функция `map()` принимает два аргумента: функцию и итерируемый объект. Функция `map()` применяет переданную функцию ко всем элементам итерируемого объекта и возвращает новый итерируемый объект с результатами. Например, можно использовать `map` для возведения каждого числа в квадрат:

```
numbers = [1, 2, 3, 4, 5]
squares = map(lambda x: x**2, numbers)
print(list(squares)) # [1, 4, 9, 16, 25]
```

Функция `filter()` также принимает два аргумента: функцию и итерируемый объект. Функция `filter()` возвращает новый итерируемый объект, содержащий только те элементы исходного итерируемого объекта, для которых переданная функция возвращает `True`. Например, можно использовать `filter()` для фильтрации только четных чисел:

```
numbers = [1, 2, 3, 4, 5]
evens = filter(lambda x: x % 2 == 0, numbers)
print(list(evens)) # [2, 4]
```

Функция `zip()` принимает два или более итерируемых объектов и возвращает итератор, который позволяет комбинировать элементы этих объектов в кортежи. Если итерируемые объекты имеют разную длину, то `zip()` вернет кортежи только для тех элементов, которые присутствуют во всех итерируемых объектах. Например:

```
numbers = [1, 2, 3]
letters = ['a', 'b', 'c']
zipped = zip(numbers, letters)
print(list(zipped)) # [(1, 'a'), (2, 'b'), (3, 'c')]
```