Skillbox

Функции map и filter

Роман Булгаков

Учитель информатики

Функция map против list comprehensions

```
animals = ['cat', 'dog', 'cow']
```

Нужно: [**'Cat'**, **'Dog'**, **'Cow'**]

Решение с map:

```
new_animals = list(map(lambda elem: elem.capitalize(), animals))
```

Решение с list comprehension:

```
new_animals = [elem.capitalize() for elem in animals]
```

1. map удобно использовать для ленивых вычислений
2. lambda замедляет map, но он всё равно **обычно** быстрее list comprehensions (если выражение достаточно простое)

Функциональное программирование

Функции **lambda**, **map**, **filter**, **zip**, а также **reduce** являются элементами функционального программирования

```python
def timer(func: Callable) -> Callable:
    """ Декоратор. Выводит время работы функции или метода """
    @functools.wraps(func)
    def wrapped(*args, **kwargs):
        start = time.time()
        result = func(*args, **kwargs)
        end = time.time()
        print('Время работы функции:', end - start)
        return result
    return wrapped
```

Это ФП

```python
@property
def age(self):
    return self.__age

@age.setter
def age(self, age):
    self.__age = age
```

И это ФП

```python
def fibonacci(number):
    cur_val = 0
    next_val = 1
    for _ in range(number):
        yield cur_val
        cur_val, next_val = next_val, cur_val + next_val
```

И это тоже ФП

Спасибо за внимание!