



Продолжение функций, кортежи

Правила тестов

- Пересдач не будет
- $0.2 / 10 = 0.02 < 0.3$
- Формула:
 $O = \text{AVG}(\text{sorted}(\text{marks}, \text{reversed}=\text{True})[:-3])$
- В конце каждого полугодия один тест на 30 минут для уверенных в себе
- Если пишете тест для уверенных в себе, то оценка будет выставлена по нему без оглядки на другие тесты

- На тест для уверенных в себе нельзя заболеть
- Тест для уверенных в себе будет приниматься по тем же правилам, что и на паре
- Тесты у всех 4 групп будут одинаковой сложности

Задание № 1

Вам дан частотный словарь слов русского языка.
Скачайте его по ссылке <https://vk.cc/a4Gkyn>
И заготовка для обработки этого словаря.

Реализуйте функцию `retrieve(filename)`. Функция должна проходить по всем строкам файла `filename` и искать в них самое частотное слово.

Не забудьте присвоить правильный путь к файлу в функции `main()`.

```
def retrieve(filename):  
    """Функция, которую нужно реализовать."""  
    pass # Помните зачем нужен оператор pass?  
  
def main():  
    filename = '<укажите путь к вашему файл>  
    most_freq_word = retrieve(filename)  
    print("Самое частотное слово:", most_freq_word)  
  
if __name__ == '__main__':  
    main()
```

Задание №1. Решение

```
def retrieve(filename):  
    most_freq_word = None  
    max_freq = -1  
    with open(filename, encoding='utf-8') as file:  
        for line in file:  
            line = line.strip()  
            word, morph, freq = line.split('|')  
            if freq > max_freq:  
                most_freq_word = word  
                max_freq = freq  
    return most_freq_word
```

Задание №2

Задание по ссылке: <https://vk.cc/a48BkX>

Решение к заданию № 2

```
def retrieve(filename, lte=None, gte=None, gender=None):
    output = []
    with open(filename, encoding='utf-8') as file:
        for line in file:
            line = line.strip() # специальный метод для очистки от всякого
            word, morph, freq = line.split('|')
            freq = float(freq)
            # Меньше либо равно
            if lte is not None and freq > lte:
                continue
            if gte is not None and freq < gte:
                continue
            if gender is not None and gender not in morph:
                continue
            output.append((word, morph, freq))
    return output
```