



работа на уроке (10.12)

1. `pymorphy`: `.word` `.lexeme` `.tag` `.cyr_repr` `.word_is_known()`

Дано два слово `'водогрязеторфопарафинолечение'`.

Проверьте, существует ли это слово в `pymorphy2`. Выведите все его теги **по-русски**, а после — склонение в формате: в одной строчке 2 слова одного падежа, разделенные 3 табуляциями (`'\t'`).

Слово	Вывод
<code>'водогрязеторфопарафинолечение'</code>	<code>False</code> СУЩ, неод, ср ед, им водогрязеторфопарафинолечение водогрязеторфопарафинолечения водогрязеторфопарафинолечения водогрязеторфопарафинолечений водогрязеторфопарафинолечению водогрязеторфопарафинолечениям водогрязеторфопарафинолечение водогрязеторфопарафинолечения водогрязеторфопарафинолечением водогрязеторфопарафинолечениями водогрязеторфопарафинолечении водогрязеторфопарафинолечениях

2. `with open(...) as ...: wget.download() .read()`

`nltk`: `.sent_tokenize()` `.word_tokenize()`

У вас есть файл с первой страницей сценария. Скачайте и разделите его на предложения и токены при помощи `nltk`. Скажите, сколько в тексте тех и других. Какие проблемы возникли?

3. `with open(...) as ...: .read()` `string.punctuation`

`nltk`: `.word_tokenize()`

Из того же самого файла удалите всю пунктуацию и еще раз посчитайте количество токенов (следите, чтобы она действительно удалилась **полностью!**). Сколько знаков препинания было в изначальном тексте?

4. `with open(...) as ...: .read()`

`nltk`: `.word_tokenize()` `stopwords.words()`

Теперь мы решили разделаться со всеми стоп-словами в сценарии. Удалите все стоп-слова и посчитайте какая доля стоп-слов была в нашем тексте (количество стоп-слов разделите на общее количество слов с ними)!

5. `dict[keys]` `if key in dict:` `.lower()`

Создайте частотный словарь со всеми токенами из сценария (советую сначала все слова привести к нижнему регистру), где ключом будет само слово, а значением — количество раз, что оно встретилось в сценарии. Выведите этот словарь.

6. `string.punctuation` `dict.fromkeys(keys, value)` `dict[key]` `key=itemgetter(1)` `sorted()`

Заведите словарь для хранения частот знаков препинания (их можно найти в модуле `string`), где изначальный ключ — это знак, а значение — ноль.

Пройдите по нашему тексту и посчитайте, сколько каких знаков препинания в нем (увеличивайте значение на 1 при встрече).

Напечатайте получившийся словарь **по убыванию частотности символов**.