**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

**іНСТИТУТ КОМП’ютерних НАУК та ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

### Кафедра “Системи автоматизованого проектування”



Звіт

до лабораторної роботи №5

на тему: “ Вивчення бібліотеки прикладних програм NLTK, для опрацювання текстів природною мовою. Початкова обробка текстів природною мовою ”

з дисципліни “Комп’ютерна лінгвістика”

Виконала:

студентка групи ПРЛм-11

Павлів І.О.

Прийняв:

викладач

Дупак Б.П.

Львів-2015

**Мета:**

* Вивчення основ програмування на мові Pathon;
* Вивчення методів роботи з файлами на локальних дисках та з Інтернету;
* Використання Юнікоду при обробці текстів;
* Нормалізація текстів, стемінг, лематизація та сегментація.

**Теоретичні відомості**

Частина електронних книжок з Project Gutenberg розповсюджується разом з NLTK у вигляді корпуса текстів. Для використання інших текстів з цього проекту можна переглянути каталог 25000 електронних книжок за адресою <http://www.gutenberg.org/catalog/> та встановити адресу (URL) потрібного текстового файлу в ASCII кодуванні. 90% текстів в Project Gutenberg є англійською мовою, але він включає також тексти більше ніж 50-ма іншими мовами (каталонська, китайська, датська, фінська, французька, німецька, італійська, португальська, іспанська…).

Виконання read() займає певний час протягом якого відбувається завантаження цієї великої книжки. Текст книжки збережений як значення змінної raw . Змінна raw містить стрічку довжиною 1,176,831 символів. (Перевірити тип змінної можна скориставшись type(raw).) Стрічка яка відповідає вмісту книжки містить багато не цікавої для аналізу інформації: пробіли, пусті стрічки, межі стрічки. Символи \r and \n , які є в тексті, це символи переводу каретки та початку нового рядка. Для подальшої роботи з текстом потрібно розділити текст на окремі слова та виділити розділові знаки. Такий процес називають токенізацією. При використанні програми токенізації з NLTK отримуємо список слів та розділових знаків.

Більшість текстів в Інтернеті є у вигляді HTML документів (файлів). Інтернет сторінки можна зберігати на диску у вигляді файлів і доступатися до них. Python також дозволяє працювати Інтернет сторінками безпосередньо використовуючи функцію urlopen.

Текст, який вивели на екран містить HTML розмітку (мета теги, JavaScript, форми , таблиці). Вилучення тексту з HTML файлу доволі розповсюджена задача, яка в NLTK вирішується за допомогою функції nltk.clean\_html(). Ця функція обробляє HTML стрічку і повертає текст у вигляді зручному для подальшої обробки (токенізації).

Для читання локальних файлів необхідно використовувати вбудовану функцію Python open() та read() метод. Якщо існує файл document.txt, то змінній raw можна присвоїти його вміст.

Інша можлива проблема при читанні текстових файлів – це різні способи маркування нового рядка у файлах різних операційних систем. При виклику функція open() може містити другий параметр для контролю відкривання файлу open('document.txt', 'rU') („r” файл для читання, ”U” universal дозволяє ігнорувати різні способи, які використовуються для маркування нового рядка).

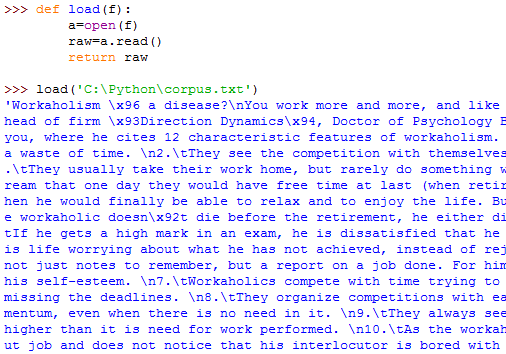
Для вводу тексту з клавіатури (при взаємодії користувача з програмою) потрібно використати функцію raw\_input().

Варіант №13

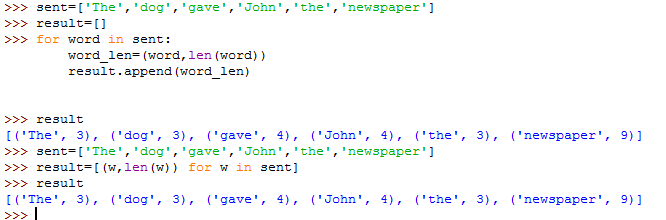
1. Напишіть функцію, яка приймає адресу URL як аргумент і повертає те, що міститься за цією адресою з видаленням HTML-розмітки. Використовувати *urllib.urlopen* для доступу до контенту наступним чином: *raw\_contents = urllib.urlopen('http://www.nltk.org/').read().*



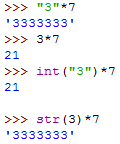
1. Збережіть деякий текст у файлі *corpus.txt*. Визначити функцію *load(f)* для читання файлу, назва якого є її аргументом і повертає стрічку, яка містить текст з файлу.



1. Перепишіть наступний цикл як *list comprehension*:

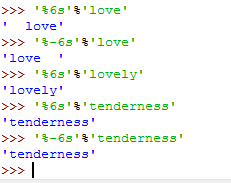


1. Перевірити різницю між стрічками і цілим виконавши наступні дії: *"3" \* 7 та 3 \* 7*. Спробуйте здійснити конвертування між стрічками і цілими використавши *int("3")* та *str(3).*



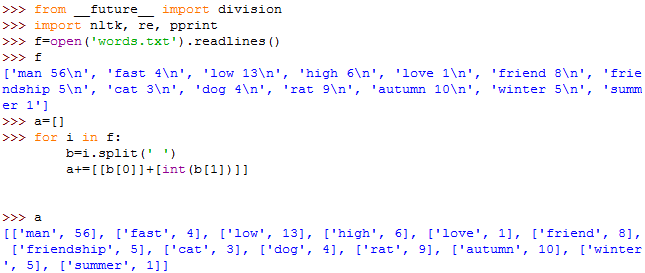
"3" – змінна, яка є стрічкового характеру, тому *"3" \* 7* отримаємо в 7 разів довшу стрічку. *int("3")* перетворює стрічкову змінну в цілочисельну, а *str(3)* перетворює цілочисельну змінну в стрічкову.

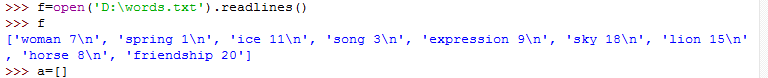
1. Що станеться, коли стрічки форматування *%6s* та *%-6s* використовуються для відображення стрічки, довшої ніж 6 символів?

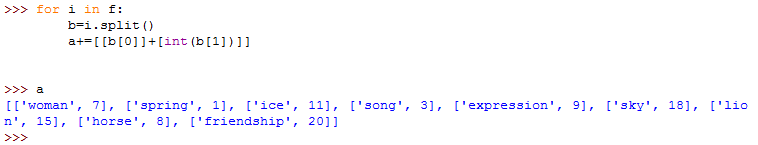


Стрічка *%6s* виконує форматування по лівому краю, а стрічка *%-6s* – форматування по правому краю. Коли ж стрічка довша за 6 символів, форматування не відбувається.

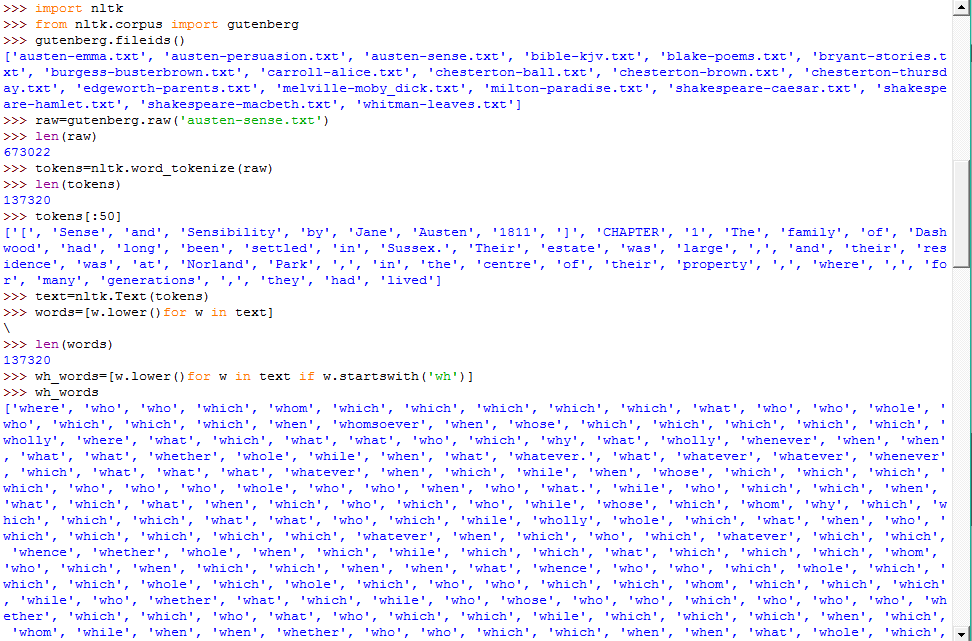
1. Створіть файл, який буде містити слова та їх частоту записані в окремих рядках через пробіл *(fuzzy 53).* Прочитайте цей файл, використовуючи *open(filename).readlines().*  Розділіть кожну стрічку на дві частини, використовуючи *split(),* і перетворіть число в ціле значення використовуючи *int().* Результат має бути у вигляді списку: *[['fuzzy', 53], ...].*



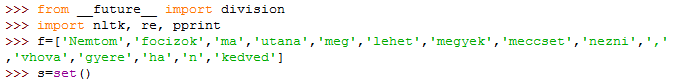


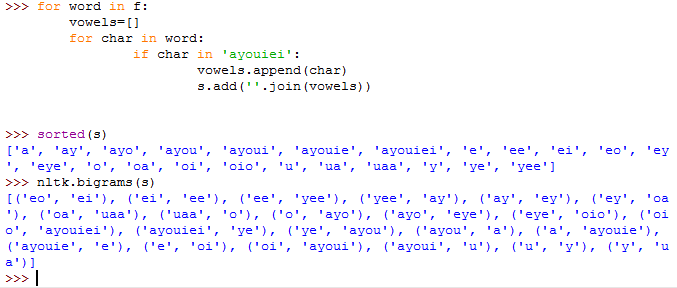


1. Прочитайте деякий текст з корпуса, здійсніть його токенізацію і збережіть у список всі wh-слова, які в ньому зустрічаються.

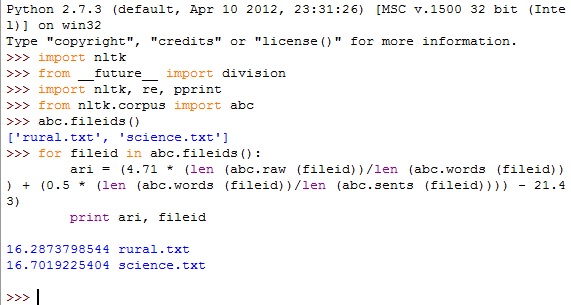


1. З тексту мовою, яка має гармонію голосних ( Hungarian), вилучіть у словах послідовності голосних і створіть частотну таблицю біграмів голосних.



****

1. Доступіться до текстів ABC Rural News та ABC Science News з корпуса (nltk.corpus.abc). Знайдіть значення для оцінки читабельності текстів (аналогічно до задачі №12). Використовуйте Punkt для поділу тексту на окремі речення.

****

Висновок:

на цій лабораторній роботі я вивчила методи роботи з файлами на локальних дисках та з Інтернету, а також методи використання Юнікоду під час обробки текстів. Також я ознайомилась з поняттями нормалізації текстів, стемінгу, лематизації та сегментації і закріпила свої знання та навички на практиці, виконавши найпоширеніші задачі.