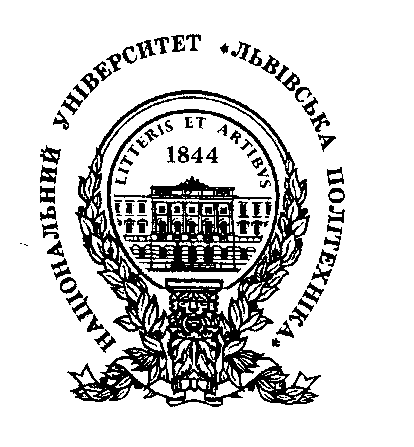
Міністерство освіти та науки України

Національний університет «Львівська Політехніка»

Інститут комп’ютерних наук та інформаційних технологій

Кафедра САПР



Лабораторна робота №5

на тему:

ВИВЧЕННЯ БІБЛІОТЕКИ ПРИКЛАДНИХ ПРОГРАМ NLTK, ДЛЯ ОПРАЦЮВАННЯ ТЕКСТІВ ПРИРОДНОЮ МОВОЮ. ПОЧАТКОВА ОБРОБКА ТЕКСТІВ ПРИРОДНОЮ МОВОЮ.

Виконала:

студентка ПРЛм-11

Михайлів Р.Б.

Перевірив:

Дупак Б.П.

Львів 2015

**МЕТА РОБОТИ**

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

**КОРОТКІ ТЕОРЕТИЧНІ ВІДОМОСТІ**

Корпуси текстів та тексти з Інтернету є важливими джерелами даних для здійснення лінгвістичних досліджень. Звичайно, якщо дослідник має власноруч зібрані тексти, то потрібні засоби для доступу до них.

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…).

Текст за номером 2554 це переклад англійською Crime and Punishment(Злочин і кара), і отримати доступ до тексту можна наступним чином:

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  | **>>> from urllib import urlopen**  **>>> url = "http://www.gutenberg.org/files/2554/2554.txt"**  **>>> raw = urlopen(url).read()**  **>>> type(raw)**  **<type 'str'>**  **>>> len(raw)**  **1176831**  **>>> raw[:75]**  **'The Project Gutenberg EBook of Crime and Punishment, by Fyodor Dostoevsky\r\n'** | |

Виконання read() займає певний час протягом якого відбувається завантаження цієї великої книжки. При використанні проксі сервера для доступу до Інтернету, при необхідності, його параметри потрібно вказати:

|  |  |
| --- | --- |
|  | **>>> proxies = {'http': 'http://www.someproxy.com:3128'}**  **>>> raw = urlopen(url, proxies=proxies).read()** |

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.

|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  | **>>> tokens = nltk.word\_tokenize(raw)**  **>>> type(tokens)**  **<type 'list'>**  **>>> len(tokens)**  **255809**  **>>> tokens[:10]**  **['The', 'Project', 'Gutenberg', 'EBook', 'of', 'Crime', 'and', 'Punishment', ',', 'by']** | |

Бібліотека NLTK використовувалась тільки на етапі токенізації і не використовувалась при доступі за адресою в Інтернеті та при читанні стрічки. Для подальшої роботи список перетворюється в NLTK текст і над ним можна здійснювати різноманітні операції:

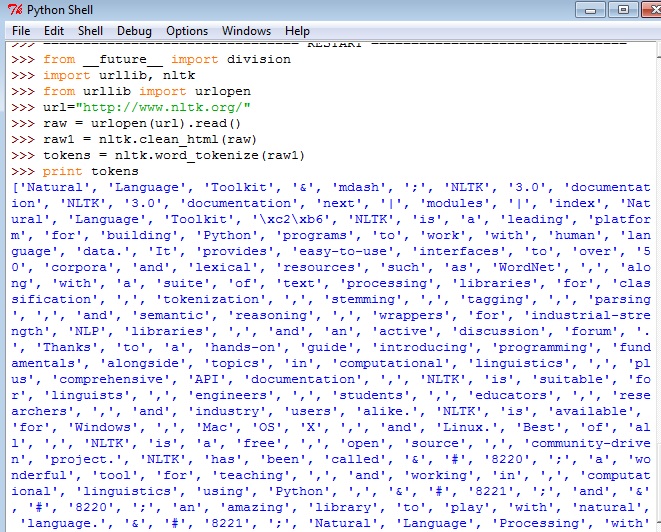
|  |  |  |
| --- | --- | --- |
| |  |  | | --- | --- | |  | **>>> text = nltk.Text(tokens)**  **>>> type(text)**  **<type 'nltk.text.Text'>**  **>>> text[1020:1060]**  **['CHAPTER', 'I', 'On', 'an', 'exceptionally', 'hot', 'evening', 'early', 'in',**  **'July', 'a', 'young', 'man', 'came', 'out', 'of', 'the', 'garret', 'in',**  **'which', 'he', 'lodged', 'in', 'S', '.', 'Place', 'and', 'walked', 'slowly',**  **',', 'as', 'though', 'in', 'hesitation', ',', 'towards', 'K', '.', 'bridge', '.']**  **>>> text.collocations()**  **Katerina Ivanovna; Pulcheria Alexandrovna; Avdotya Romanovna; Pyotr**  **Petrovitch; Project Gutenberg; Marfa Petrovna; Rodion Romanovitch;**  **Sofya Semyonovna; Nikodim Fomitch; did not; Hay Market; Andrey**  **Semyonovitch; old woman; Literary Archive; Dmitri Prokofitch; great**  **deal; United States; Praskovya Pavlovna; Porfiry Petrovitch; ear rings** | |

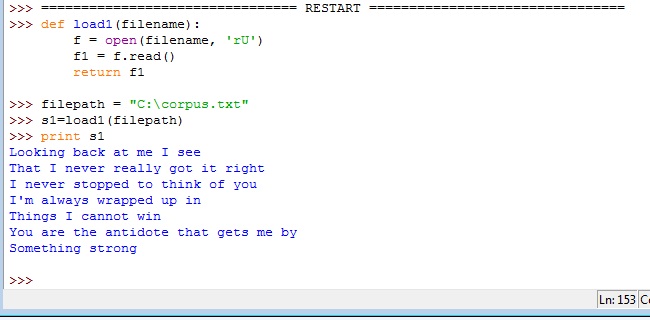
В побудованих колокаціях зустрічається Project Gutenberg, словосполучення, яке не міститься в тексті книжки. Завантажений текст з сайту містить метатекстову розмітку (інформацію про автора, про текст, про людей які готували електронний варіант та ін.). Ця інформація може бути, як на початку тексту так і в його кінці . Для роботи власне з текстом книжки потрібно в ручному режимі знайти межі цих додаткових даних і за допомогою зрізів доступитися до тексту.

**ТЕКСТИ ПРОГРАМ НА МОВІ PYTHON**

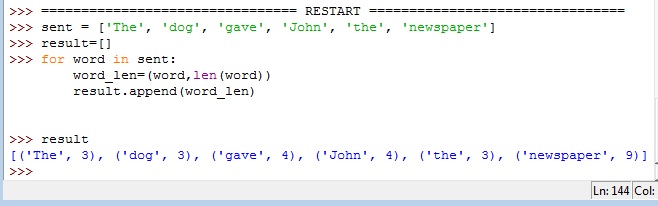
**Варіант 20**

5.1. 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().

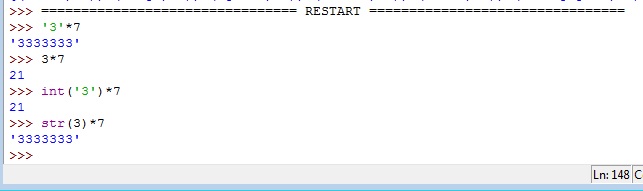


5.2. Збережіть деякий текст у файлі corpus.txt. Визначити функцію load(f) для читання файлу, назва якого є її аргументом і повертає стрічку, яка містить текст з файлу.

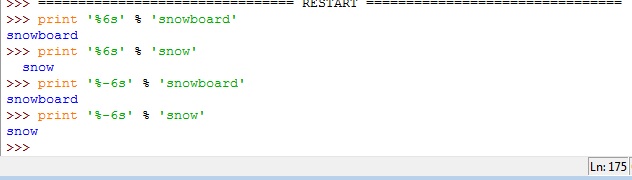
5.3. Перепишіть наступний цикл як list comprehension:



5.4. 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).

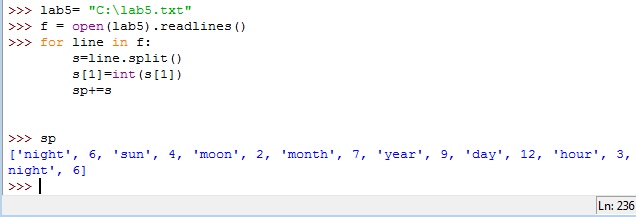


5.5. Що станеться, коли стрічки форматування %6s та %-6s використовується для відображення стрічки довшої ніж 6 символів?

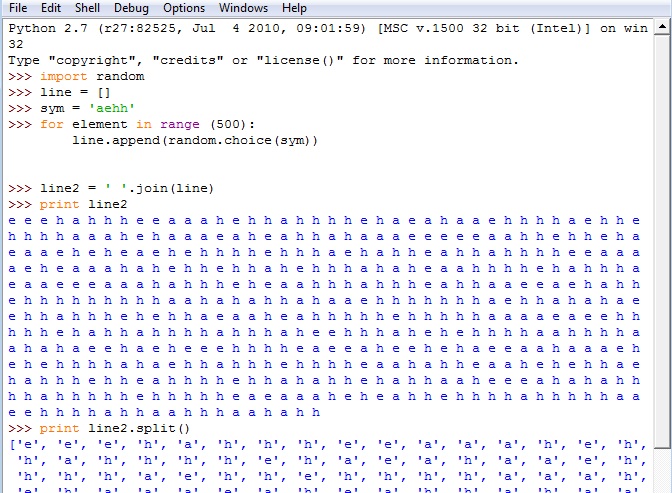


Стрічка %6s для стрічок довжиною до шести символів виконує форматування по лівому краю (тобто додає пробіли), а стрічка %-6s – форматування по правому краю. Коли ж стрічка довша за 6 символів, форматування не відбувається.

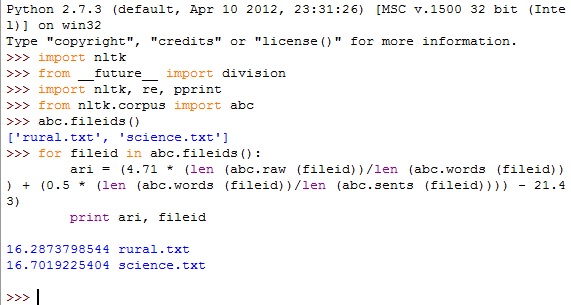
5.7. Створіть файл, який буде містити слова та їх частоту записані в окремих рядках через пробіл ( fuzzy 53). Прочитайте цей файл використовуючи open(filename).readlines(). Розділіть кожну стрічку на дві частини використовуючи split(), і перетворіть число в ціле значення використовуючи int(). Результат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



5.10. Модуль random включає функцію choice(), яка випадковим чином вибирає елементи послідовності. Наприклад, choice("aehh ") буде вибирати один з чотирьох символів. Напишіть програму генерації стрічки з 500 випадково вибраних символів "aehh ". Для поєднання елементів в стрічку використовуйте ''.join() . Нормалізуйте отриманий результат використовуючи split() та join().



5.14. 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, ми вивчили основи програмування на мові Python, методи роботи з файлами на локальних дисках, використання Юнікоду при обробці текстів, нормалізації текстів, стемінгу, лематизації та сегментації.