**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”**

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

### Кафедра “Системи автоматизованого проектування”



Звіт

до лабораторної роботи №6

на тему: “ ВИВЧЕННЯ БІБЛІОТЕКИ ПРИКЛАДНИХ ПРОГРАМ NLTK, ДЛЯ ОПРАЦЮВАННЯ ТЕКСТІВ ПРИРОДНОЮ МОВОЮ. ВИКОРИСТАННЯ РЕГУЛЯРНИХ ВИРАЗІВ ДЛЯ ОБРОБКИ ТЕКСТУ ”

з дисципліни “Комп’ютерна лінгвістика”

Виконала:

студентка групи ПРЛм-11

Павлів І.О.

Прийняв:

викладач

Дупак Б.П.

Львів-2015

**Мета роботи:**

* вивчення основ програмування на мові *Python*;
* використання регулярних виразів для обробки текстів.

**Короткі теоретичні відомості**

Багато задач лінгвістичних досліджень передбачають встановлення відповідності заданому шаблону. Наприклад, можна знайти слова, які закінчуються на “ed” використовуючи метод endswith('ed'). Подібні методи перевірки слів перелічені в Таблиці Методичних вказівок до лабораторної роботи №2. Регулярні вирази є більш потужним і гнучким методом опису шаблонів символів, які необхідно виявити у послідовностях символів. **Регулярний вираз (вислів)** (в [програмуванні](http://uk.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F)) — це [рядок](http://uk.wikipedia.org/wiki/%D0%A0%D1%8F%D0%B4%D0%BE%D0%BA_%28%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D1%83%D0%B2%D0%B0%D0%BD%D0%BD%D1%8F%29) що описує або збігається з [множиною](http://uk.wikipedia.org/wiki/%D0%9C%D0%BD%D0%BE%D0%B6%D0%B8%D0%BD%D0%B0) рядків, відповідно до набору спеціальних [синтаксичних](http://uk.wikipedia.org/wiki/%D0%A1%D0%B8%D0%BD%D1%82%D0%B0%D0%BA%D1%81%D0%B8%D1%81) правил. Регулярні вислови використовуються в багатьох [текстових редакторах](http://uk.wikipedia.org/wiki/%D0%A2%D0%B5%D0%BA%D1%81%D1%82%D0%BE%D0%B2%D0%B8%D0%B9_%D1%80%D0%B5%D0%B4%D0%B0%D0%BA%D1%82%D0%BE%D1%80) та допоміжних інструментах для пошуку та зміни тексту на основі заданих шаблонів.

Для роботи з регулярними виразами у Python потрібно імпортувати бібліотеку re скориставшись: import re. Корпус слів англійської мови Words Corpus буде використовуватися в якості лінгвістичних даних серед яких буде проводитися пошук.

Синтаксис регулярних висловів залежить від інтерпретатора, що використовується для їх обробки. Однак, із незначними відхиленнями, майже всі поширені механізми інтерпретатори регулярних висловів мають спільні правила.

Найпростіший регулярний вислів, з якого формуються складні, є звичайний символ. Більшість символів, включаючи усі літери та цифри, є регулярними висловами, що співпадають із відповідними символами в рядках.

Пошук слів із закінченням ed можна здійснити використовуючи регулярний вираз «ed$». Потрібно використати функцію re.search(p, s), яка перевіряє чи може зразок p бути знайдений у будь-якому місці стрічки s. Потрібно визначити символи, які шукаємо та використати символ долара , який в регулярних виразах позначає кінець слова.

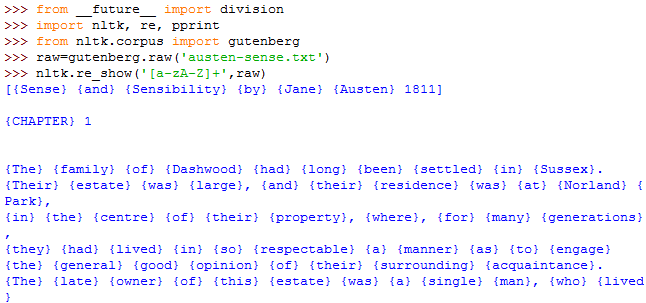
Символ ”.” універсальний символ , якому відповідає будь-який один символ. Нехай потрібно знайти слова з восьми літер, де j – третя літера та t – шоста літера. При створенні регулярного виразу у місцях де може бути будь-який символ вказується крапка. Символ ”^” вказує на початок стрічки.

Оператор ^ виконує іншу функцію, якщо його записати першим символом в квадратних дужках. Наприклад вираз «[^aeiouAEIOU]» встановлює відповідність до будь-яких символів крім голосних. В корпусі NPS Chat Corpus можна знайти , за допомогою виразу «^[^aeiouAEIOU]+$» всі слова в яких повністю відсутні голосні : :):):), grrr, cyb3r , zzzzzzzz. Також сюди увійшли слова які містять інші символи крім букв.

Символ зворотної похилої риски (backslash) означає що наступний метасимвол втрачає своє спеціальне. Отже \. означає просто крапку. Послідовність у фігурних дужках, подібна до {3,5}, вказує на кількість повторів. Вертикальна риска вказує на можливість вибору між елементами, які стоять справа і зліва від неї. Круглі дужки вказують на область дії оператора. Вони можуть використовуватися з вертикальною рискою (д**из'юнкція**), подібно до виразу «w(i|e|ai|oo)t», який встановлює відповідність до слів wit, wet, wait, та woot.

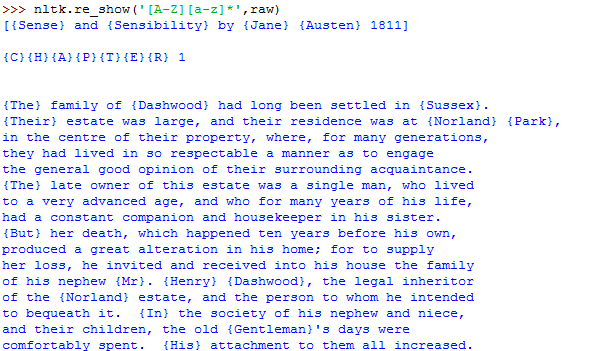
**Варіант №13**

1. Описати, які класи стрічок відповідають наступному регулярному виразу. [a-zA-Z]+. Результати перевірити використовуючи nltk.re\_show()



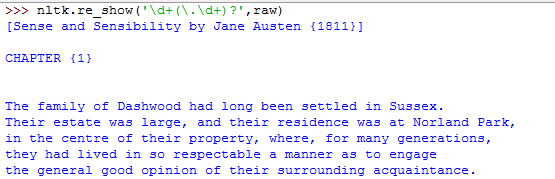
Цьому виразу відповідають стрічки довільної довжини, які складаються з великих та (або) малих літер.

1. Описати, які класи стрічок відповідають наступному регулярному виразу. [A-Z][a-z]\*. Результати перевірити використовуючи nltk.re\_show()



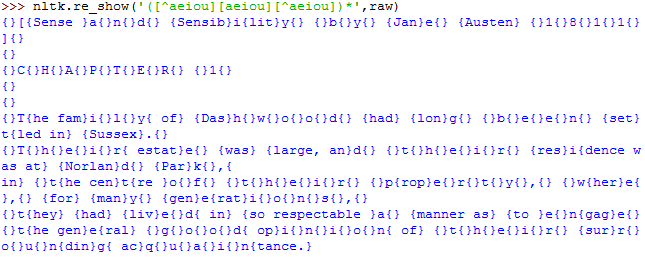
Цьому виразу відповідають стрічки, які складаються з однієї великої літери та 0 або більше малих літер.

1. Описати, які класи стрічок відповідають наступному регулярному виразу. \d+(\.\d+)?. Результати перевірити використовуючи nltk.re\_show()



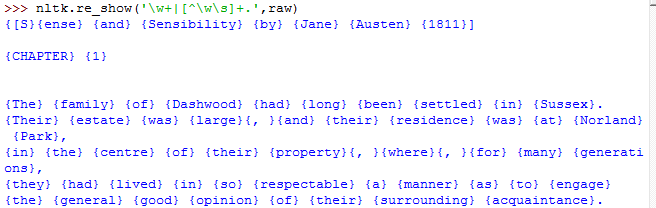
Даному виразу відповідають усі можливі додатні числа.

1. Описати, які класи стрічок відповідають наступному регулярному виразу. ([^aeiou][aeiou][^aeiou])\*. Результати перевірити використовуючи nltk.re\_show()



Цьому виразу відповідають 0 або більше послідовностей не голосна-голосна-не голосна.

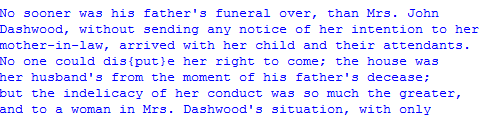
1. Описати, які класи стрічок відповідають наступному регулярному виразу. \w+|[^\w\s]+.. Результати перевірити використовуючи nltk.re\_show()



Виразу \w+|[^\w\s]+. відповідають стрічки, які складаються з довільної к-сті букв або цифр (\w+), або з довільної к-сті не букв, цифр, пробілів ([^\w\s]+) і будь-якого символу (.).

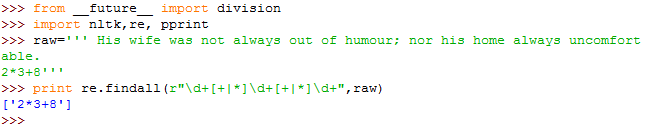
1. Описати, які класи стрічок відповідають наступному регулярному виразу. p[aeiou]{,2}t Результати перевірити використовуючи nltk.re\_show()



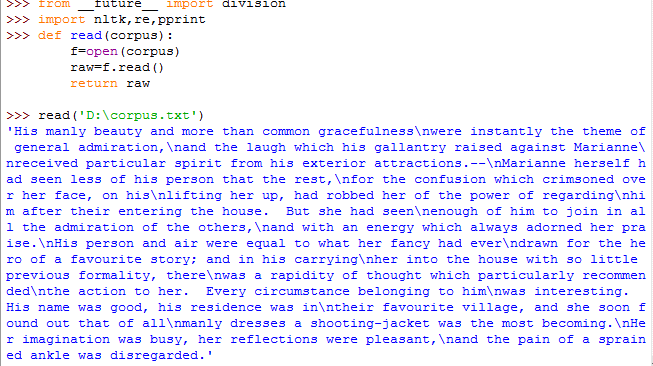


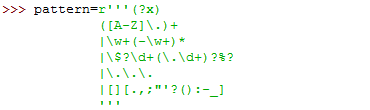
Цьому виразу відповідають стрічки, які складаються з букви ‘p’, нуля, одної або двох голосних і букви ‘t’.

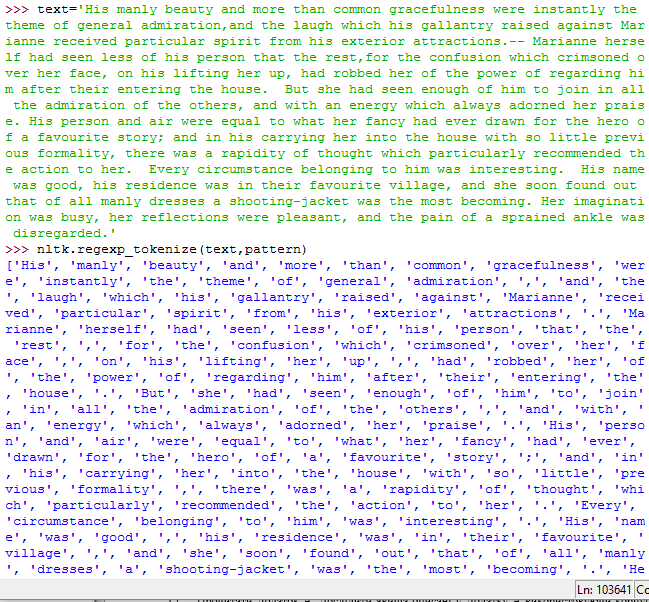
1. Написати регулярний вираз, який встановлює відповідність наступному класу стрічок: арифметичний вираз з цілими значеннями і, який містить операції множення та додавання *(2\*3+8*).



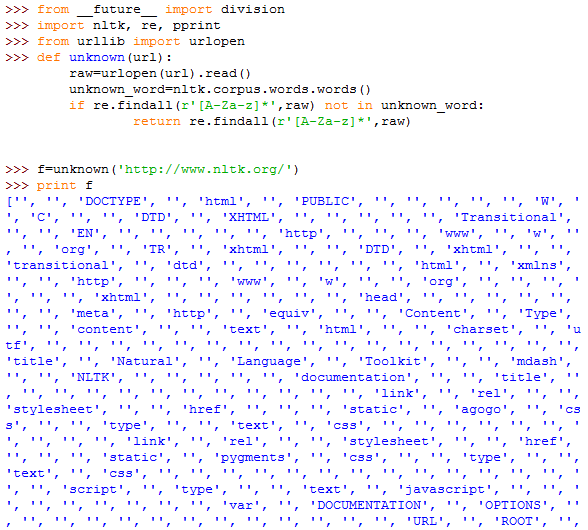
1. Зберегти довільний текст у файлі corpus.txt. Визначити функцію для читання з цього файлу (назва файлу аргумент функції) і повертає стрічку, яка містить текст з файлу. Використовуючи nltk.regexp\_tokenize() розробити токенізатор для токенізації різних типів пунктуації в цьому тексті. Використовувати багаторядковий запис регулярного виразу з коментарями та «verbose flag»



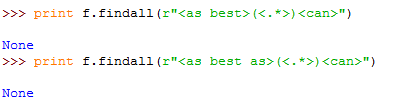
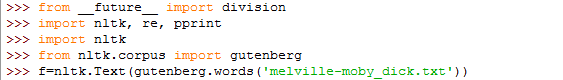


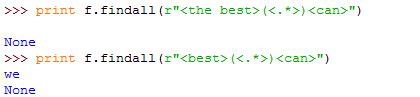


1. Написати функцію unknown(), яка приймає інтернет адресу як аргумент і повертає не відомі слова, які зустрічаються в тексті. При розробці функції використовувати re.findall() для виявлення всіх підстрічок та корпус Words Corpus (nltk.corpus.words) для виявлення не відомих слів.



1. Прочитати Додаток А. Дослідити явища описані у Додатку А використовуючи корпуси текстів та метод findall()для пошуку в токенізованому тексті.





**Висновки:**

на цій лабораторній роботі я розглянула поняття регулярних виразів та навчилась з ними працювати. Дізналась про те, що таке метасимволи, токенізатор та розглянула основні метасимволи регулярних виразів. Навчилась обробляти фрагменти слів.