МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"
ІНСТИТУТ КОМП'ЮТЕРНИХ НАУК ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра "Системи автоматизованого проектування"

Звіт

до лабораторної роботи №2

на тему: "ВИВЧЕННЯ БІБЛІОТЕКИ ПРИКЛАДНИХ ПРОГРАМ NLTK, ДЛЯ ОПРАЦЮВАННЯ ТЕКСТІВ ПРИРОДНОЮ МОВОЮ.
ОСНОВИ ПРОГРАМУВАННЯ НА МОВІ PYTHON(частина 2) "

з дисципліни "Комп'ютерна лінгвістика"

Виконала:
студентка групи ПРЛм-11
Гарбуз Л.В.
Прийняв:
викладач
Дупак Б.П.

Львів 2015

Мета роботи: вивчити основи програмування на мові *Python,* ознайомитись з контрольними структурами та класом FreqDist.

Теоретичні відомості.

Python підтримує широкий набір операторів для встановлення взаємозв'язків між змінними (значеннями). Повний набір цих операторів наведений у таблиці 1.

Таблиця 1.

Operator	Relationship
<	less than
<=	less than or equal to
==	equal to (note this is two not one = sign)
!=	not equal to
>	greater than
>=	greater than or equal to

Загальна схема роботи цих прикладів ([w for w in text if *condition*]), де *condition* умова, яка справджується або ні (приймає значення True або False).

Звичайно ми використовуємо умовні оператори, як частину *If* операторів. Для перевірки властивостей окремих слів існує набір наступних функцій (Таблиця2.).

Функція	Пояснення
s.startswith(t)	чи починається s з t
s.endswith(t)	чи закінчується s на t
t in s	Чи t міститься в s
s.islower()	Чи всі символи в s є малі
s.isupper()	Чи всі символи в s є великі
s.isalpha()	Чи всі символи в s є букви
s.isalnum()	Чи всі символи в s є букви і цифри
s.isdigit()	Чи всі символи в s є цифри
s.istitle()	Чи всі слова в s є з великої літери

В даних прикладах наступні вирази: [f(w) for ...] or [w.f() for ...], де f це функція, яка або визначає довжину слова або перетворює малі літери на великі. В кожному з цих прикладів здійснюється обробка кожного елемента списку. Змінній W послідовно присвоюються значення слів з текста і над цією змінною виконуються передбачені програмою дії. Такий запис [f(w) for ...] називається "list comprehension." (включення списків або спискові висловлювання) і є важливим для написання та розуміння програм на Python.

Для автоматичного визначення слів, які є найбільш інформативними для текстів певного жанру або певної тематики спочатку інтуітивно виникає думка побудувати частотний список або частотний розподіл. Частотний розподіл вказує на частоту з якою в тексті зустрічається кожне зі слів. Такий частотний список називають розподілом тому, що він вказує яким чином загальна кількість слів розподіляється між словниковими статями (оригінальні слова) в тексті. Враховуючи що побудова частотних розподілів часто необхідна при обробці природної мови в NLTK реалізовано окремий клас FreqDist в модулі nltk.probability . Застосуємо цей клас для знаходження 50 найчастотніших слів в тексті *Moby Dick.*

Тексти програм на мові *Python*.

Варіант – 3

3.3. Створіть змінну sentence і присвойте їй значення ‘she sells sea shells by the sea shore’ та напишіть фрагмент програми, яка генерує нову стрічку додаючи ‘like’ перед кожним зі слів , яке починається з ‘se’.

```
sentence = 'she sells sea shells by the sea shore'
words = sentence.split()
s=''
for word in words:
    if word.startswith('se'):
        s = s +' '+'like'+' '+ word
    else:
        s = s + ' ' + word
print s
>>>
 she like sells like sea shells by the like sea shore
...
```

3.5. Пуста стрічка і пустий список в частині умов if виразу призводить до помилки. Напишіть програму для демонстрації таких випадків при використанні if тверджень.

```
>>> str = []
>>> for element in str :
        print element


>>> str = ['hello', 'bye']
>>> for element in str :
        if:

SyntaxError: invalid syntax
```

3.8. Виконати наступні приклади і пояснити різницю між ними w.isupper() not w.islower()

```
>>> w = ('LIDIIA')
>>> w.isupper ()
True
>>> not w.islower ()
True
```

3.9. Знайдіть в тексті № 5 всі слова довжина яких дорівнює 4 і побудуйте для них частотний розподіл.

```
>>> import nltk
>>> from nltk.book import*
*** Introductory Examples for the NLTK Book ***
Loading text1, ..., text9 and sent1, ..., sent9
Type the name of the text or sentence to view it.
Type: 'texts()' or 'sents()' to list the materials.
text1: Moby Dick by Herman Melville 1851
text2: Sense and Sensibility by Jane Austen 1811
text3: The Book of Genesis
text4: Inaugural Address Corpus
text5: Chat Corpus
text6: Monty Python and the Holy Grail
text7: Wall Street Journal
text8: Personals Corpus
text9: The Man Who Was Thursday by G . K . Chesterton 1908
>>> fdist5 = FreqDist (text5)
>>> sorted ([w for w in set (text5) if len (w) ==4])
[u'!!!!', u'!!!.', u'!...', u'!???', u'"...', u'####', u'((((', u'))))', u',,,,', u'.. .', u'....', u'.op.',
>>> fdist5= FreqDist (text5)
>>> fdist5
<FreqDist with 6066 samples and 45010 outcomes>
>>> vocabulary1 = fdist5.keys ()
>>> vocabulary1 [:50]
[u'.', u'JOIN', u'PART', u'?', u'lol', u'to', u'i', u'the', u'you', u',', u'I', u'a', u'hi', u'me', u'...',
>>> fdist5= FreqDist ()
KeyboardInterrupt
>>> fdist5= FreqDist (sorted ([w for w in set (text5) if len (w) ==4]))
>>> fdist5
<FreqDist with 1181 samples and 1181 outcomes>
>>> vocabulary1 = fdist5.keys ()
>>> vocabulary1 [:50]
[u'!!!!', u'!!!.', u'!...', u'!???', u'"...', u'####', u'((((', u'))))', u',,,,', u'.. .', u'....', u'.op.',
```

3.11. Напишіть вираз для знаходження в тексті №6 всіх слів які відповідають наступним вимогам: закінчуються на ize; містять літеру z; містять послідовність літер pt; написані з великої літери . Результат представити, як список слів.

```
>>> fdist6= FreqDist (text6)
>>> fdist6
<FreqDist with 2166 samples and 16967 outcomes>
>>> fdist6= FreqDist (text6)
>>> sorted ([w for w in set (text6) if w.endswith('ize') or 'z' in w or 'pt' in w or w.istitle()])
[u'A', u'Aaaaaaaaah', u'Aaaaaaaah', u'Aaaaaah', u'Aaaah', u'Aaaaugh', u'Aaagh', u'Aaah', u'Aaauggh', u'Aaaugh',
```

3.16. Побудуйте колокації для текстів №1 та №5. Результати порівняйте.

```
import nltk
from nltk.book import*
text1.collocations()
text5.collocations()
a = []
a.append(text1.collocations)
b = []
b.append(text5.collocations)
if (b==a or a==b):
    print 'Collocations are identical'
else:
        print 'Collocations are not identical'
```

```
>>> ================================ RESTART ================================
>>>
*** Introductory Examples for the NLTK Book ***
Loading text1, ..., text9 and sent1, ..., sent9
Type the name of the text or sentence to view it.
Type: 'texts()' or 'sents()' to list the materials.
text1: Moby Dick by Herman Melville 1851
text2: Sense and Sensibility by Jane Austen 1811
text3: The Book of Genesis
text4: Inaugural Address Corpus
text5: Chat Corpus
text6: Monty Python and the Holy Grail
text7: Wall Street Journal
text8: Personals Corpus
text9: The Man Who Was Thursday by G . K . Chesterton 1908
Building collocations list
Sperm Whale; Moby Dick; White Whale; old man; Captain Ahab; sperm
whale; Right Whale; Captain Peleg; New Bedford; Cape Horn; cried Ahab;
years ago; lower jaw; never mind; Father Mapple; cried Stubb; chief
mate; white whale; ivory leg; one hand
Building collocations list
wanna chat; PART JOIN; MODE #14-19teens; JOIN PART; cute.-ass MP3; MP3
player; times .. .; ACTION watches; guys wanna; song lasts; last
night; ACTION sits; -...)...- S.M.R.; Lime Player; Player 12%; dont
know; lez gurls; long time; gently kisses; Last seen
Collocations are not identical
```

Висновок: на лабораторній роботі №2 я вивчила основи програмування на мові *Python*, ознайомилась з контрольними структурами та класом FreqDist,а також з операторами та новими функціями.