

МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
КАФЕДРА МО ЭВМ

ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №1
ПО ДИСЦИПЛИНЕ «ИНФОРМАТИКА»
Тема: Основные управляющие конструкции. Wikipedia API

Студентка гр. 0382

Деткова А.С.

Преподаватель

Шевская Н.В.

Санкт-Петербург

2020

Цель работы.

Изучить основы языка Python, циклы while, for, условный оператор if, принцип ООП (объектно-ориентированное программирование), функции, модули.

Научиться использовать модуль Wikipedia API.

Задание.

Используя вышеописанные инструменты, напишите программу, которая принимает на вход строку вида

название_страницы_1, название страницы_2, ... название_страницы_n,
сокращенная_форма_языка

и делает следующее:

1. Проверяет, есть ли такой язык в возможных языках сервиса, если нет, выводит строку "no results" и завершает выполнение программы. В случае, если язык есть, устанавливает его как язык запросов в текущей программе.

2. Ищет максимальное число слов в кратком содержании страниц "название_страницы_1", "название страницы_2", ... "название_страницы_n", выводит на экран это максимальное количество и название страницы (т.е. её **title**), у которой оно обнаружилось. Считается, что слова разделены пробельными символами.

Если максимальных значений несколько, выведите последнее.

3. Строит список-цепочку из страниц и выводит полученный список на экран.

Элементы списка-цепочки - это страницы "название_страницы_1", "название страницы_2", ... "название_страницы_n", между которыми может быть одна промежуточная страница или не быть промежуточных страниц.

Предположим, нам на вход поступила строка:

Айсберг, IBM, ru

В числе ссылок страницы с названием "Айсберг", есть страница с названием , которая содержит ссылку на страницу с названием "Буран", у которой есть ссылка на страницу с названием "IBM" -- это и есть цепочка с промежуточным звеном в виде страницы "Буран".

Гарантируется, что существует или одна промежуточная страница или ноль: т.е. в числе ссылок первой страницы можно обнаружить вторую.

Цепочка должна быть кратчайшей, т.е. если существуют две цепочки, одна из которых содержит промежуточную страницу, а вторая нет, стройте цепочку без промежуточного элемента.

Пример входных данных:

Айсберг, IBM, ru

Пример вывода:

115 IBM

['Айсберг', 'Буран', 'IBM']

Первая строка содержит решение подзадачи №2, вторая - №3.

Основные теоретические положения.

Модуль Wikipedia API — модуль, который позволяет программно работать с wiki-страницами.

Функции:

1. `wikipedia.page(title)` — поиск страницы с названием `title`, представляет собой объект класса `WikipediaPage`, который представляет собой страничку сервиса `Wikipedia`.
2. `wikipedia.languages()` - поиск всех возможных языков сервиса, возвращает словарь, ключами которого являются сокращенные названия языков, значениями — названия.
3. `wikipedia.set_lang(lang)` - установить язык `lang`, как язык запросов в текущей программе.

Поля класса:

1. `page.summary` — строка, в которой содержится краткое содержание страницы `page`.
2. `page.title` — строка, в которой содержится название страницы `page`.
3. `page.links` — список, который содержит названия страниц, ссылки на которые содержит страница `page`.

Выполнение работы.

В программе для решения поставленной задачи используется 3 функции, каждая из которых выполняет какую-то задачу. Плюс используется `is_page_valid(page)`, которая возвращает *True*, если страница с названием `page` найдена и *False* — если не найдена.

Функция `lang(s)`: аргументом является `s` — строка, сокращенное название языка, переданное пользователем; функция ищет среди всех возможных языков `wikipedia` (метод `wikipedia.languages()`) введенный язык и если он найден, то возвращает саму строку `s`, если нет, то строку *'no results'*.

Функция `max__quantity_of_words(List)`: аргумент — список `list`, в который передаются все названия страниц введенные пользователем. Функция находит максимальное количество слов в кратком содержании страницы и ее имя. Используются методы: `str.split()` - делит строку `str` по символу заключенному в скобках, возвращает список элементов, используется для деления строки на слова и их подсчета. Возвращает список, в котором 0 элемент — максимальное количество слов в описании страницы, а 1ый — её заголовок.

Функция `list_res(list)`: аргумент — список `list`, в который передаются все названия страниц введенные пользователем. Функция возвращает список-цепочку, каждый элемент, которого это страницы, между которыми может быть одна промежуточная страница или не быть промежуточных страниц.

Этот список кратчайший. Используется метод *List.append()* для класса список — добавляется элемент из скобок в конец списка *List*. Возвращает список *res*.

На вход от пользователя поступает строка из слов, разделенных запятой и пробелом, которая преобразуется в список *L* с помощью метода *split()*.

Далее вызывается функция *lang(L[len(L)-1])*, в качестве аргумента передается последний элемент списка *L* — краткое название языка.

Если функция вернула значение *'no results'* — выводится это сообщение, иначе: вызываются функции *max__quantity_of_words(L[0:len(L)-1:])* и *,list_res(L[0:len(L)-1:])* аргумент которых — срез списка *L* от нулевого до предпоследнего индекса (без введенного языка) и выводят на экран результаты работы этих функций.

Разработанный программный код см. в приложении А.

Тестирование.

Результаты тестирования представлены в табл. 1.

Таблица 1 – Результаты тестирования

№ п/п	Входные данные	Выходные данные	Комментарии
1.	Айсберг, IBM, ru	115 IBM [„Айсберг“, „Буран“, „IBM“]	Страница „Айсберг“ не содержит прямой ссылки на „IBM“, поэтому идет через страницу „Буран“.
2.	Кошка, сиамская кошка, ugab	no results	Нет языка ugab в wikipedia.
3.	cat, UNIX, en	344 Cat [„Cat“, „UNIX“]	Страница „cat“ напрямую содержит „UNIX“.

Выводы.

Было изучены основы языка Python, циклы while, for, условный оператор if, принцип ООП (объектно-ориентированное программирование), функции, модули.

Было исследовано, как использовать модуль Wikipedia API.

Разработана программа, выполняющая считывание с клавиатуры строки с названиями страниц, перечисленных через запятую и пробел, и сокращенное название языка. Для обработки данных использовались функции *lang(s)*, *max__quantity_of_words(List)*, *list_res(list)*. Если введенный язык существует, то он устанавливался как язык запросов и вызывались другие функции, которые выводили максимальное количество слов в кратком содержании страницы и ее названии и список-цепочку, если язык не существует, то выводилось сообщение 'no results'. Во избежание возникновения исключительных ситуаций был использована функция *is_page_valid(page)*.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД ПРОГРАММЫ

Название файла: lb1.py

```
import wikipedia

def is_page_valid(page):
    try:
        wikipedia.page(page)
    except Exception:
        return False
    return True

def lang(s):
    d=list((wikipedia.languages()).keys())
    for i in d:
        if s==i:
            return s
    return 'no results'

def max__quantity_of_words(List):
    quantity = 0
    name = ''
    for i in range(len(List)):
        if is_page_valid(List[i]) == True:
            page=wikipedia.page(List[i])
            if len((page.summary).split()) >= quantity:
                quantity = len((page.summary).split())
                name = page.title
    res = [quantity, name]
    return res

def list_res(list):
    res = []
    k = 0
    while is_page_valid(list[k]) == False:
        k += 1
    res.append(list[k])
    for k in range(len(list)-1):
        flag = 0
        if is_page_valid(list[k + 1]) == True:
            pages_links = (wikipedia.page(list[k])).links
            if list[k + 1] in pages_links:
                res.append(list[k + 1])
                flag = 1
        if flag == 0:
            for j in range(len(pages_links)):
                if flag == 0:
                    pages2_links = (wikipedia.page(pages_links[j])).links
                    for i in range(len(pages2_links)):
                        if list[k + 1] == pages2_links[i]:
                            res.append(pages_links[j])
                            res.append(list[k + 1])
                            flag = 1
```

```

        else:
            k += 1
        return res

L = input().split(' ', ' ')

res = lang(L[len(L)-1])
if res == 'no results':
    print(res)
else:
    wikipedia.set_lang(res)
    print(max__quantity_of_words(L[0:len(L) - 1:])[0],
max__quantity_of_words(L[0:len(L) - 1:])[1])
    print(list_res(L[0:len(L) - 1:]))

```