16-2-22:c05

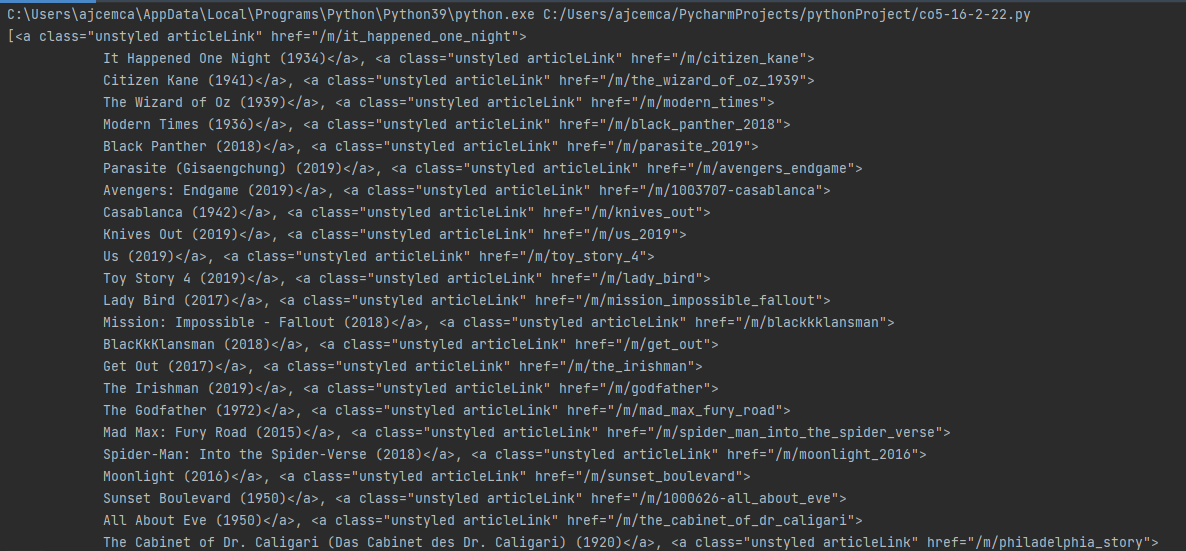
Aim: implement a simple web crawler(ensure ethical conduct).

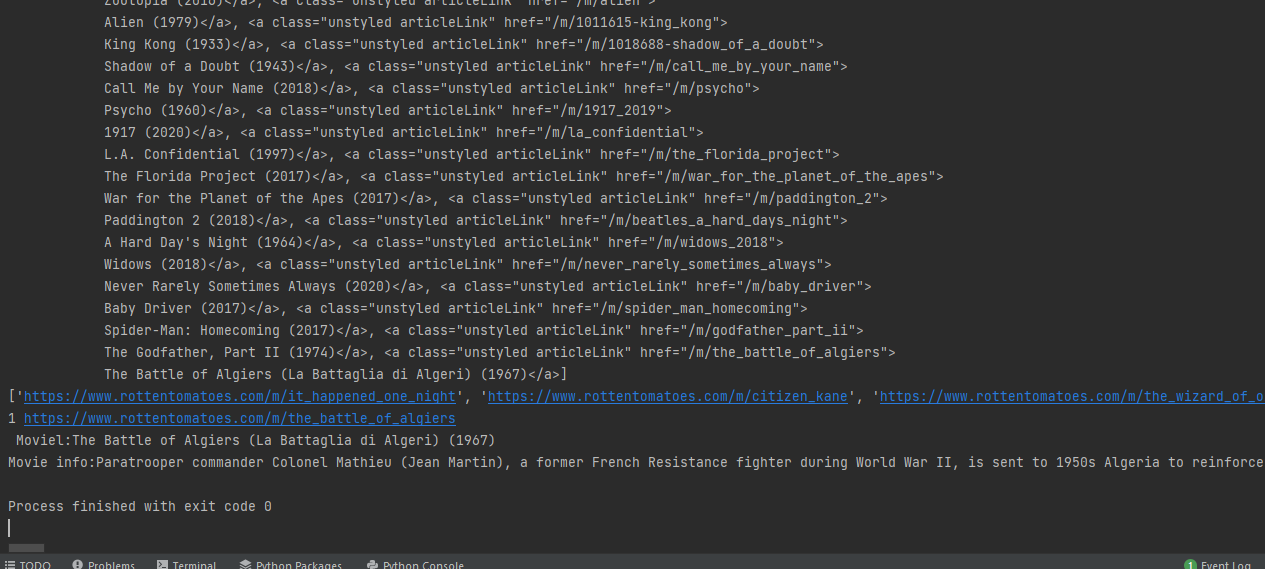
I

**Program**

import requests  
import lxml  
from bs4 import BeautifulSoup  
url = "https://www.rottentomatoes.com/top/bestofrt/"  
headers = {  
 'User-Agent' :'Mozilla/5.0(Windows NT 6.1;WOW64) AppleWebkit/537.36 (KHTML, like Geck) Chrome/63.0.3239,132safari/537.36QIHU 360SE'  
  
}  
f = requests.get(url, headers = headers)  
movies\_list=[]  
soup= BeautifulSoup(f.content, 'html.parser')  
movies = soup.find('table', {  
 'class': 'table'  
}) .find\_all('a')  
print(movies)  
num = 0  
for anchor in movies:  
 urls = 'https://www.rottentomatoes.com' +anchor['href']  
 movies\_list.append(urls)  
print(movies\_list)  
num +=1  
movie\_url = urls  
movie\_f = requests.get(movie\_url, headers = headers)  
movie\_soup = BeautifulSoup(movie\_f.content, 'lxml')  
movie\_content= movie\_soup.find('div',{  
 'class':'movie\_synopsis clamp clamp-6 js-clamp'  
})  
print(num, urls,'\n', 'Movie:' +anchor.string.strip())  
print('Movie info:' + movie\_content.string.strip())

**Output**



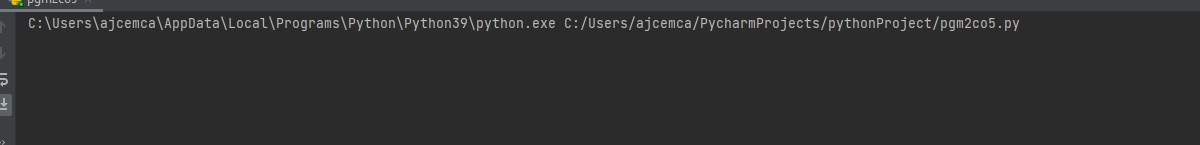


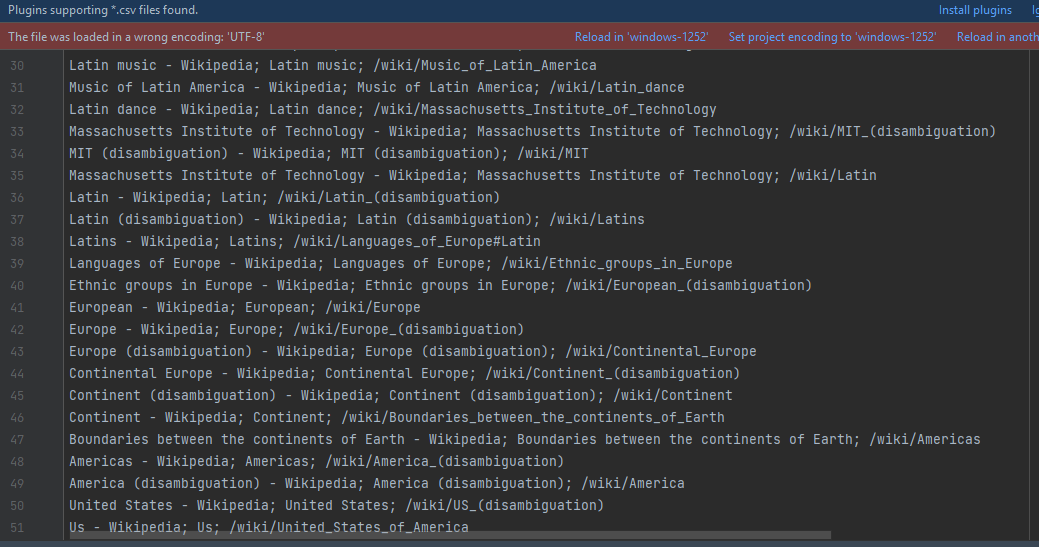
**Program 2:**

**Aim;**

import requests  
  
from bs4 import BeautifulSoup  
pages\_crawled = []  
  
def crawler(url):  
 page = requests.get(url)  
 soup =BeautifulSoup(page.text, 'html.parser')  
 links =soup.find\_all('a')  
  
  
 for link in links:  
 if 'href' in link.attrs:  
 if link['href'].startswith('/wiki') and ':' not in link['href']:  
 if link['href'] not in pages\_crawled:  
 new\_link = f"https://en.wikipedia.org{link['href']}"  
 pages\_crawled.append(link['href'])  
 try:  
 with open('data.csv','a') as file:  
 file.write(f'{soup.title.text}; {soup.h1.text}; {link["href"]}\n')  
 crawler(new\_link)  
  
 except:  
 continue  
  
crawler('https://en.wikipedia.org')

**Output**



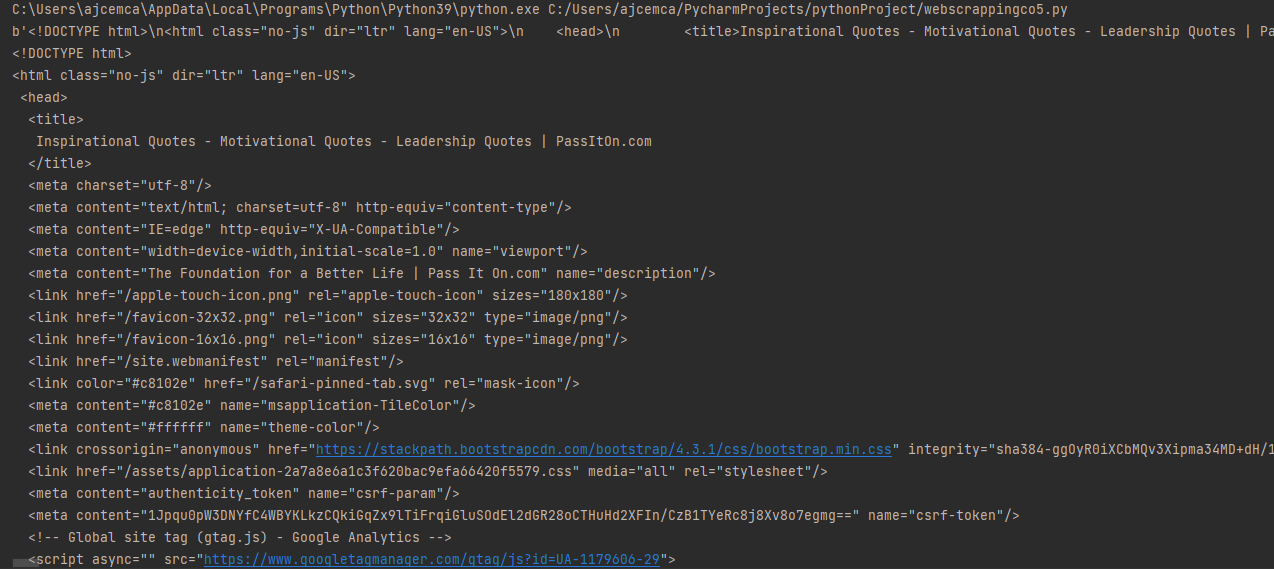


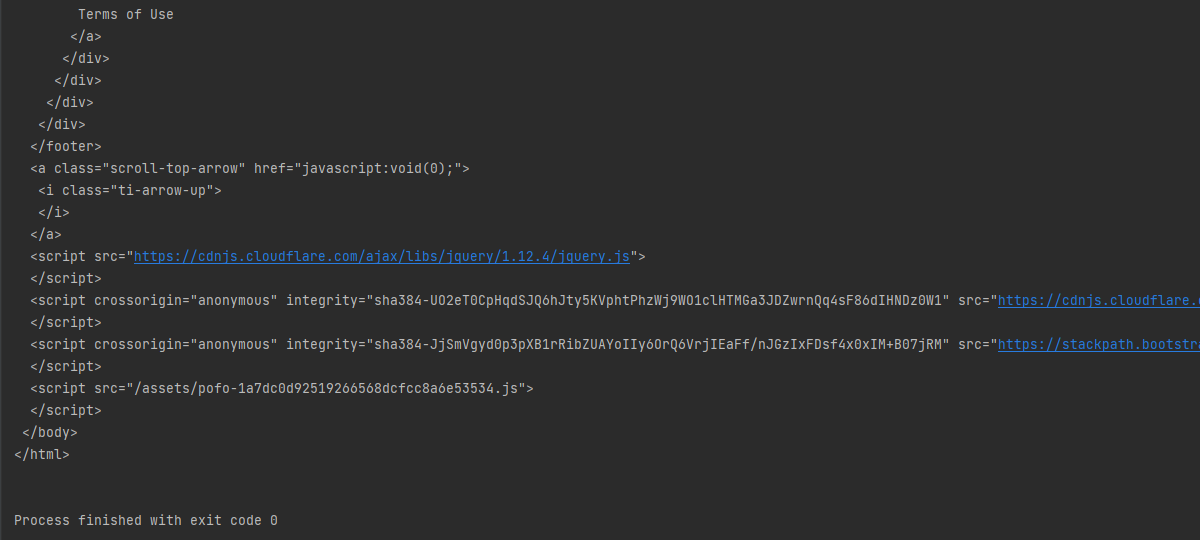
**PROGRAM**

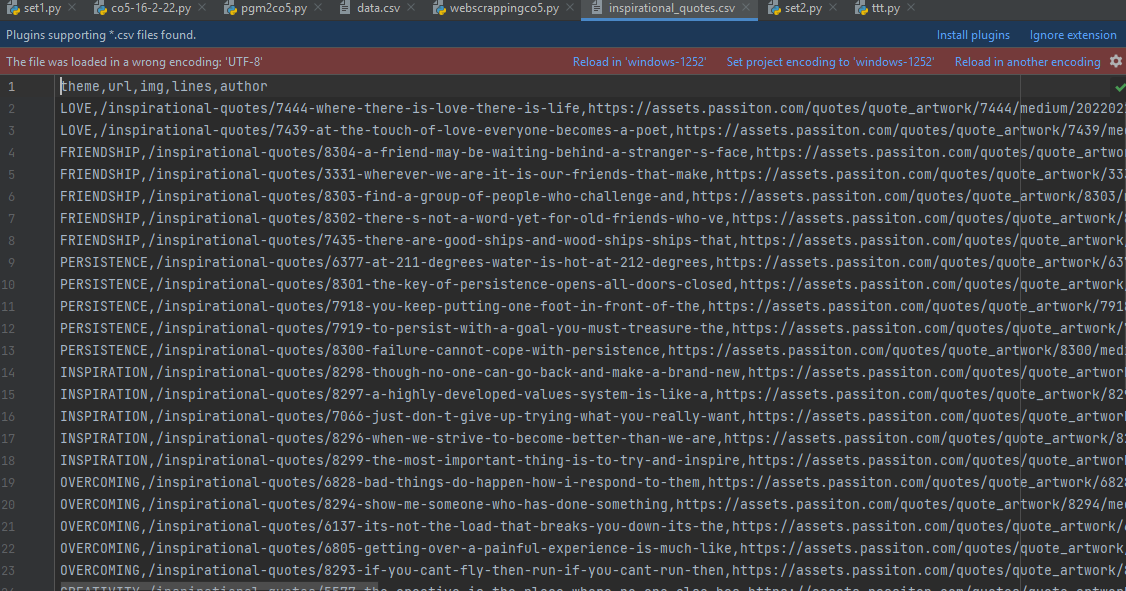
**Aim: implement a program to scarp the webpage of any popular website?**

import requests  
from bs4 import BeautifulSoup  
import csv  
import lxml  
URL = "https://www.values.com/inspirational-quotes"  
r = requests.get(URL)  
print(r.content)  
soup = BeautifulSoup(r.content, 'lxml')  
print(soup.prettify())  
quotes = []  
table = soup.find('div', attrs={'id': 'all\_quotes'})  
for row in table.findAll('div',attrs={'class':'col-6 col-lg-3 text-center margin-30px-bottom sm-margin-30px-top'}):  
 quote = {}  
 quote['theme'] = row.h5.text  
 quote['url'] = row.a['href']  
 quote['img'] = row.img['src']  
 quote['lines'] = row.img['alt'].split("#")[0]  
 quote['author'] = row.img['alt'].split("#")[1]  
 quotes.append(quote)  
filename = 'inspirational\_quotes.csv'  
with open(filename, 'w', newline='') as f:  
 w = csv.DictWriter(f, ['theme', 'url', 'img', 'lines', 'author'])  
 w.writeheader()  
 for quote in quotes:  
 w.writerow(quote)

**Output**



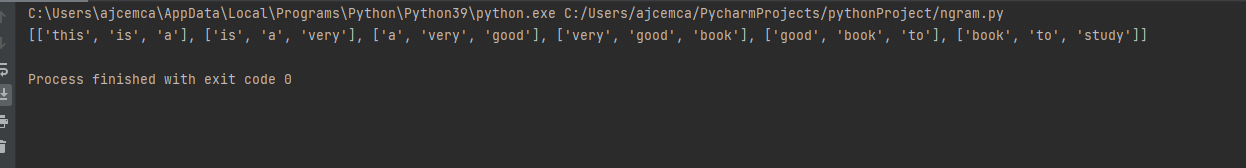




**Program 4:ngram**

def generate\_ngrams(text, WordsToCombine):  
 words = text.split()  
 output = []  
 for i in range(len(words) - WordsToCombine + 1):  
 output.append(words[i:i + WordsToCombine])  
 return output  
x=generate\_ngrams(text='this is a very good book to study', WordsToCombine=3)  
print(x)

**Output**



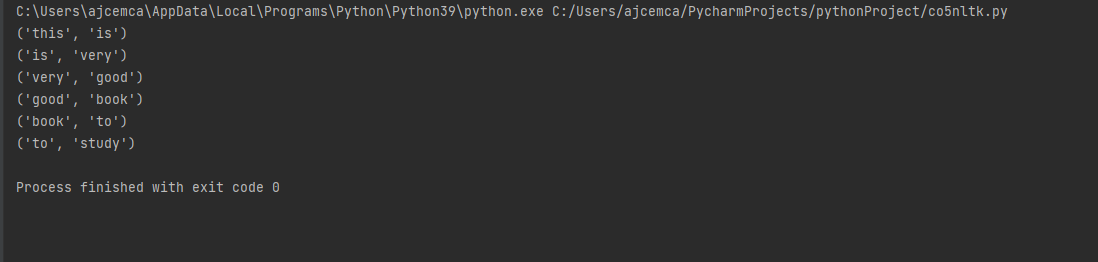
**Program 5**

**Aim:**

**Python program for natural language processing NGRAM(without using inbuilt function)**

import nltk  
from nltk.util import ngrams  
  
sampleText = 'this is very good book to study'  
NGRAMS = ngrams(sequence=nltk.word\_tokenize(sampleText), n=2)  
for grams in NGRAMS:  
 print(gram

**output**



**PROGRAM:6**

**AIM: python program for natural language processing - speech tagging.**

import nltk  
nltk.download  
from nltk .corpus import stopwords  
from nltk.tokenize import word\_tokenize, sent\_tokenize  
  
stop\_words = set(stopwords.words('english'))  
  
#dummy text  
txt = "Sukany, Rajib and Naba are my good friends. "\  
 "sukanya getting married"\  
 "sunday holiday and i love it. "\  
 "Marriage is big step in one's life."\  
 "it is both exiting and frightening. "\  
 "but friendship is a scared bond between people. "\  
 "it is a special kind of love between us."\  
 "many of you must have tried searching for a friend. "\  
 "but never found the right one. "  
  
tokenize = sent\_tokenize(txt)  
for i in tokenize:  
 wordsList = nltk.word\_tokenize(i)  
 wordsList = [w for w in wordsList if not w in stop\_words]  
  
 tagged= nltk.pos\_tag(wordsList)  
 print(tagged)

**output**

