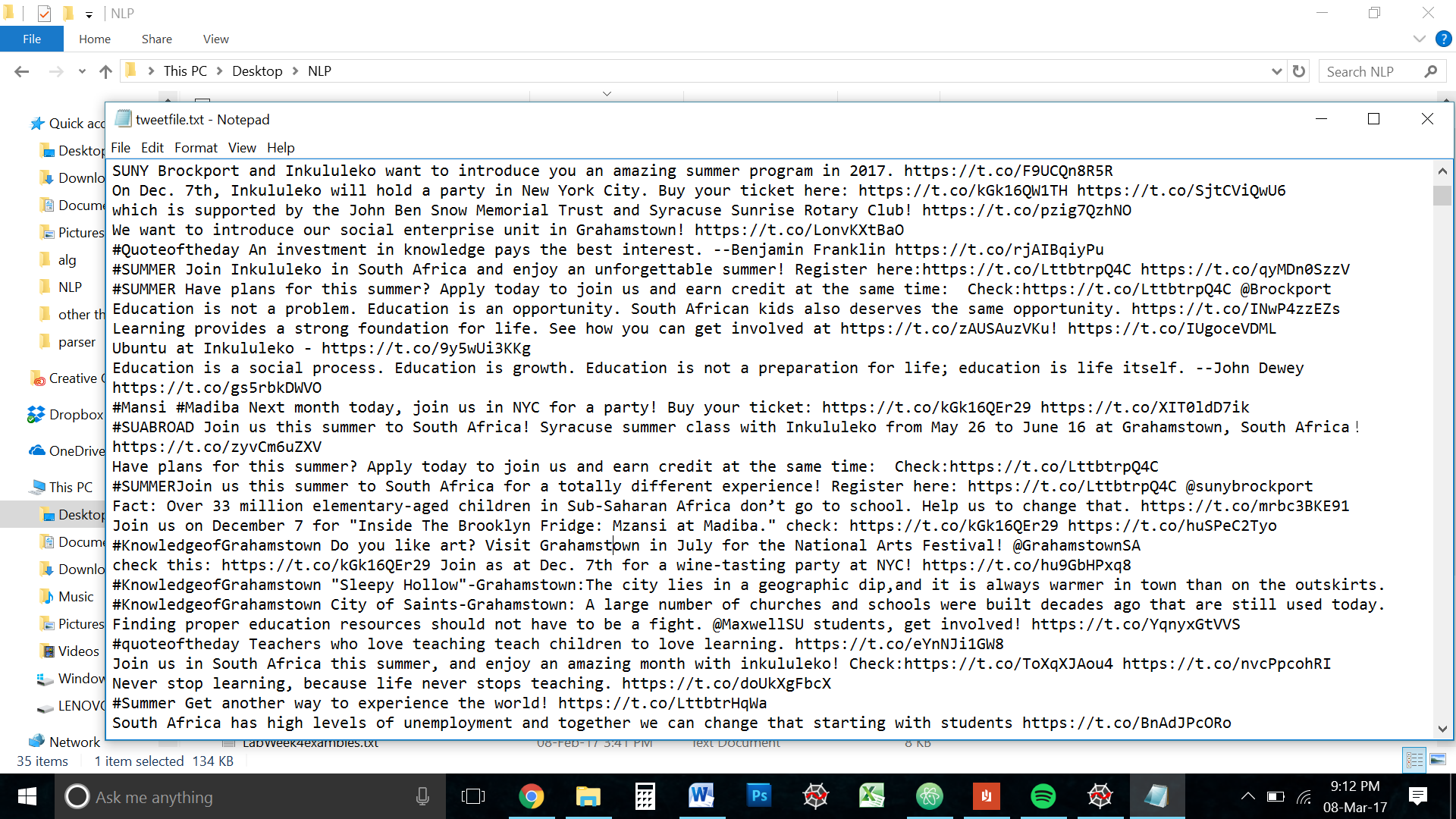
**CIS 668: Natural Language Processing**

**Assignment:2 Mahima Singh 454995544**

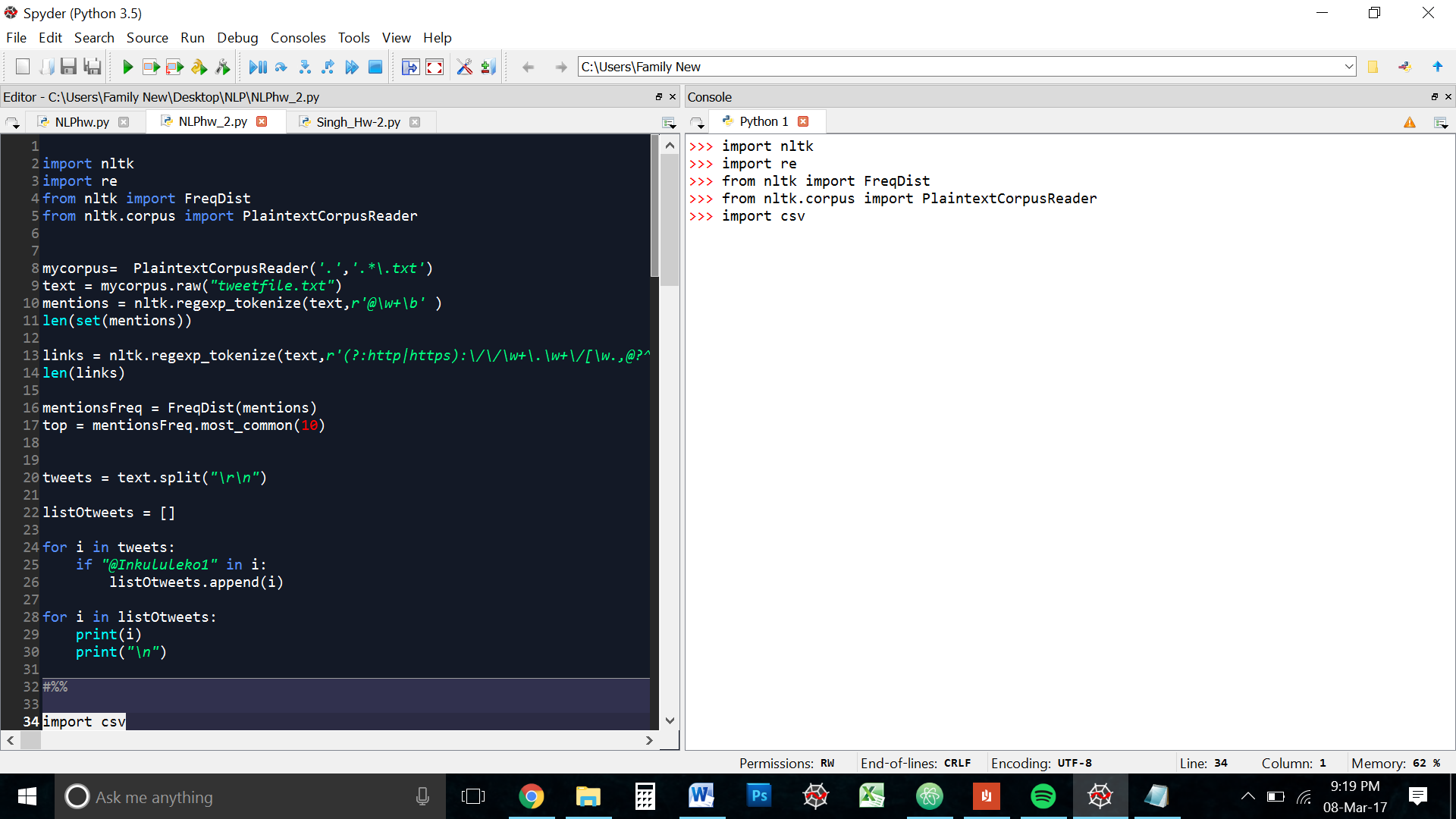
1. **Using Regular Expressions to Analyze Twitter Data**

I first looked at the text file in a text editor to get a sense of the data.



Upon preliminary analysis, it is was clear that the test is straight forward and not very complicated. Quick, keyword searches confirmed that there were many twitter accounts that were mentioned multiple times.

After a quick import of all the necessary libraries



I ran a test on the text file to understand the EOL characters and how the tweets were separated from each other. 

I quickly noticed that the tweets were separated by the characters “\r\n”

Keeping this in mind I continued my investigation.

* 1. **How many twitter accounts were mentioned with @?**

Just to be sure I did a keyword search on the doc with “@”

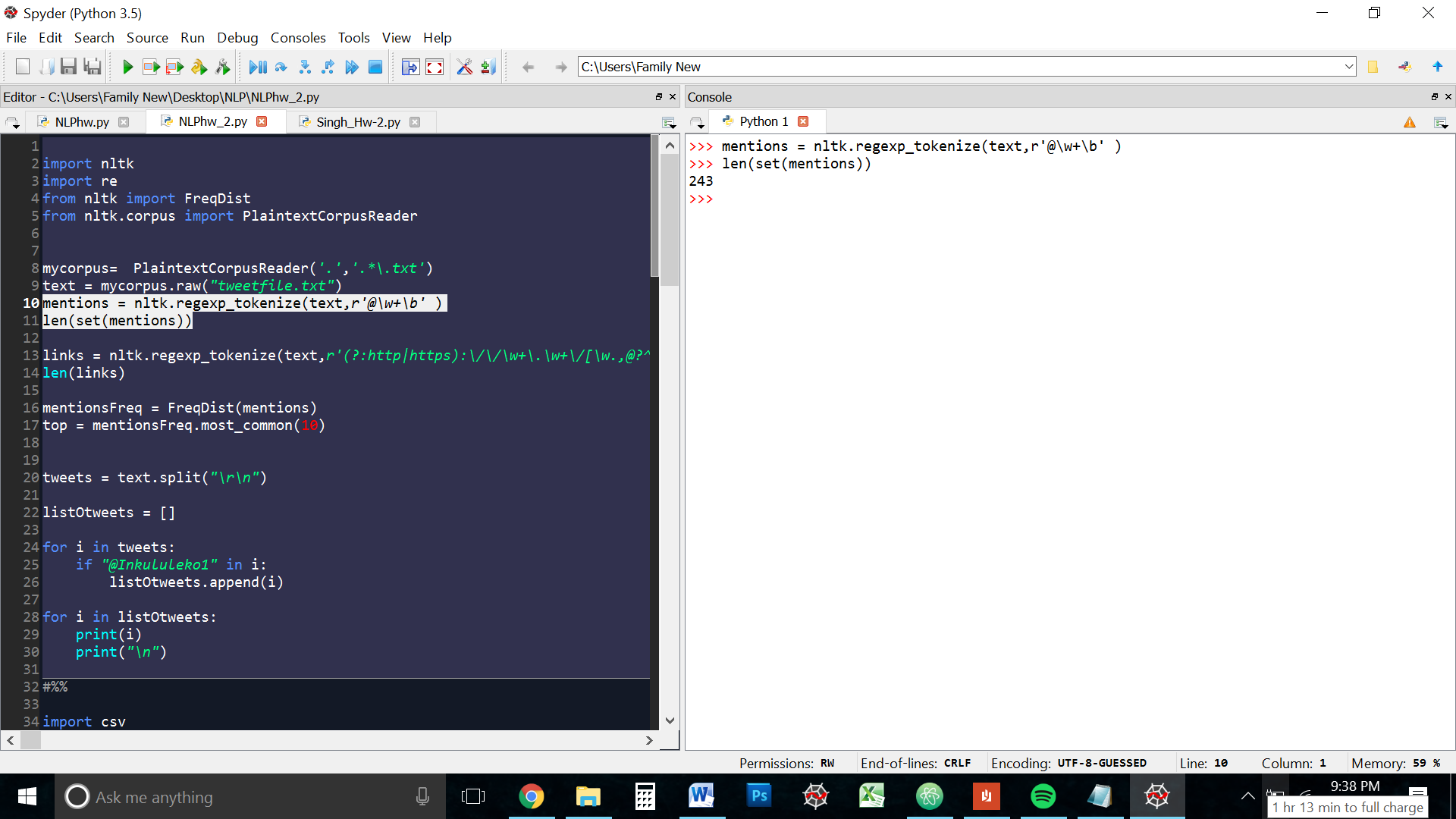
It turned up 570 results. With that as my starting point, I began experimenting with various RegEx. I use the Atom text editor because it allows for inline RegEx Search. The answer was simple enough r\@\w+b\ But to be sure I also tried the capital ‘W’With the Regex r\@\W+b\ the search turned up six results.

@@

@

and other single ‘@’ signs. These weren’t what the question is asking for. So I decided to stick with the r\@\w+b\ expression after trying the expression r\@[\w!#$%^&\*()-]+b\ which yielded the same results. So I chose the smaller expression for simplicity.

I then ran the expression through a RegEx tokenizer on the text and stored it in a list called ‘mentions’



As you can see, with my chosen regular expression, the number of unique accounts mentioned in the text is **243**

Excerpt of ‘mentions’:

@inkululeko1

@SUAbroad

@Inkululeko1

@Jason\_Torreano

@inkululeko1

@suabroad

@SUNYEmpire

@Inkululeko1

@globalyouthcnct

@Jason\_Torreano

@gruber\_law

@Jason\_Torreano

@Inkululeko1

@inkululeko1

@inkululeko1

@Inkululeko1

@Inkululeko1

@Inkululeko

@Inkululeko1

@Heifer

@Inkululeko1

@SyracuseU

@SUNYEmpireALB

@StudentDotCom

@inkululeko

@daviscollege

@devryuniv

@dowlingcollege

@CIMtweet

@daemencollege

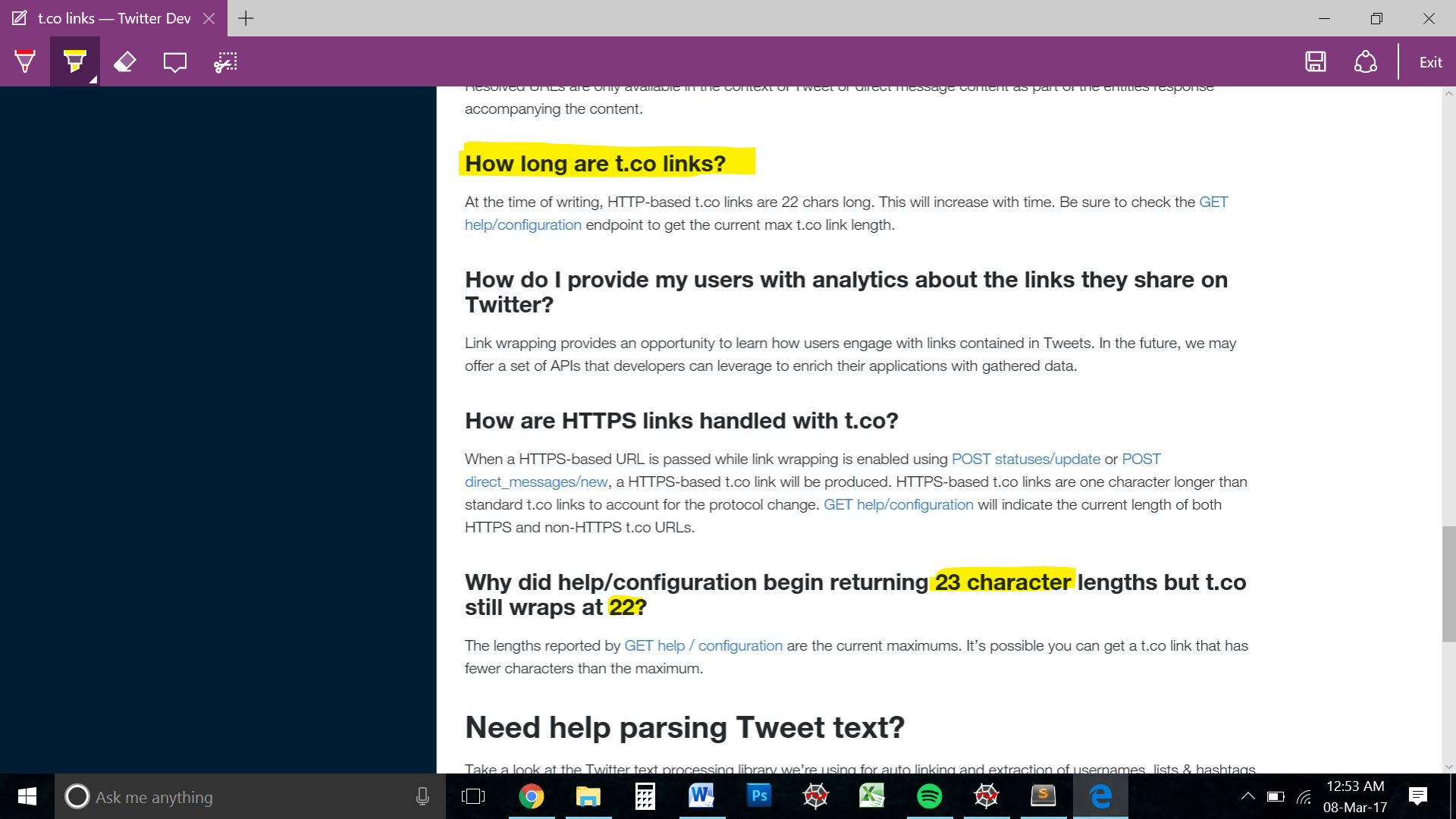
@CNR1904

@CollegeofStRose

@ColumbiaUniver

* 1. **How many web links were provided in the tweets?**

This was an interesting RegEx to write. All the links on twitter are t.co links. I did a little [research](https://dev.twitter.com/basics/tco) on them and found a few patterns



But this turns out to be untrue. I limited the Regex to match only 10 characters after https|http://t.co/ But some tweets towards the end (newer tweets) have only 6 to 8 characters after the t.co element.

I ran many versions of this RegEx to find the best match

**(http|https):\/\/(\w+\.\w+\/)+(\w\*[\w@?^=%&\/~+#-])?**

**(http|https):\/\/(\w+\.\w+)([\w.,@?^=%&:\/~+#-]\*[\w@?^=%&\/~+#-])?**

**(http|https):\/\/(\w+\.\w+)([\w.,@?^=%&:\/~+#-]\*[\w@?^=%&\/~+#-])?(?=\s)**

**(http|https):\/\/(\w+\.\w+\/)([\w.,@?^=%&:\/~+#-]\*)**

**(http|https):\/\/(\w+\.\w+\/)([\w.,@?^=%&:\/~+#-]\*)(?=\s)**

**(?:http|https):\/\/\w+\.\w+\/[\w.,@?^=%&:\/~+#-]\*(?=\s)**

Grouping was ticky. And the multiple backslashes were vconfusing. But once I got used to them, it as smooth sailing.

*Below is a breakdown of the final Regex for this question*

***(?:http|https):\/\/\w+\.\w+\/[\w.,@?^=%&:\/~+#-]\*(?=\s)***

***(?:http|https)***

*To match http or https. The (?:\_\_\_) is a non-grouping selection. This way the https/http doest get separated from the whole selection when I run the RegEx through the tokenizer.*

***:\/\/***

*This is matches the ://*

***\w+\.\w+\/***

*Matches t.co/ the first \w+ matches the ‘t’. I could have matched it with a single \w as well because I knew the links were going to follow this format. But I decided to keep it safe and use the +sign to indicate one re more matches.*

***[\w.,@?^=%&:\/~+#-]\****

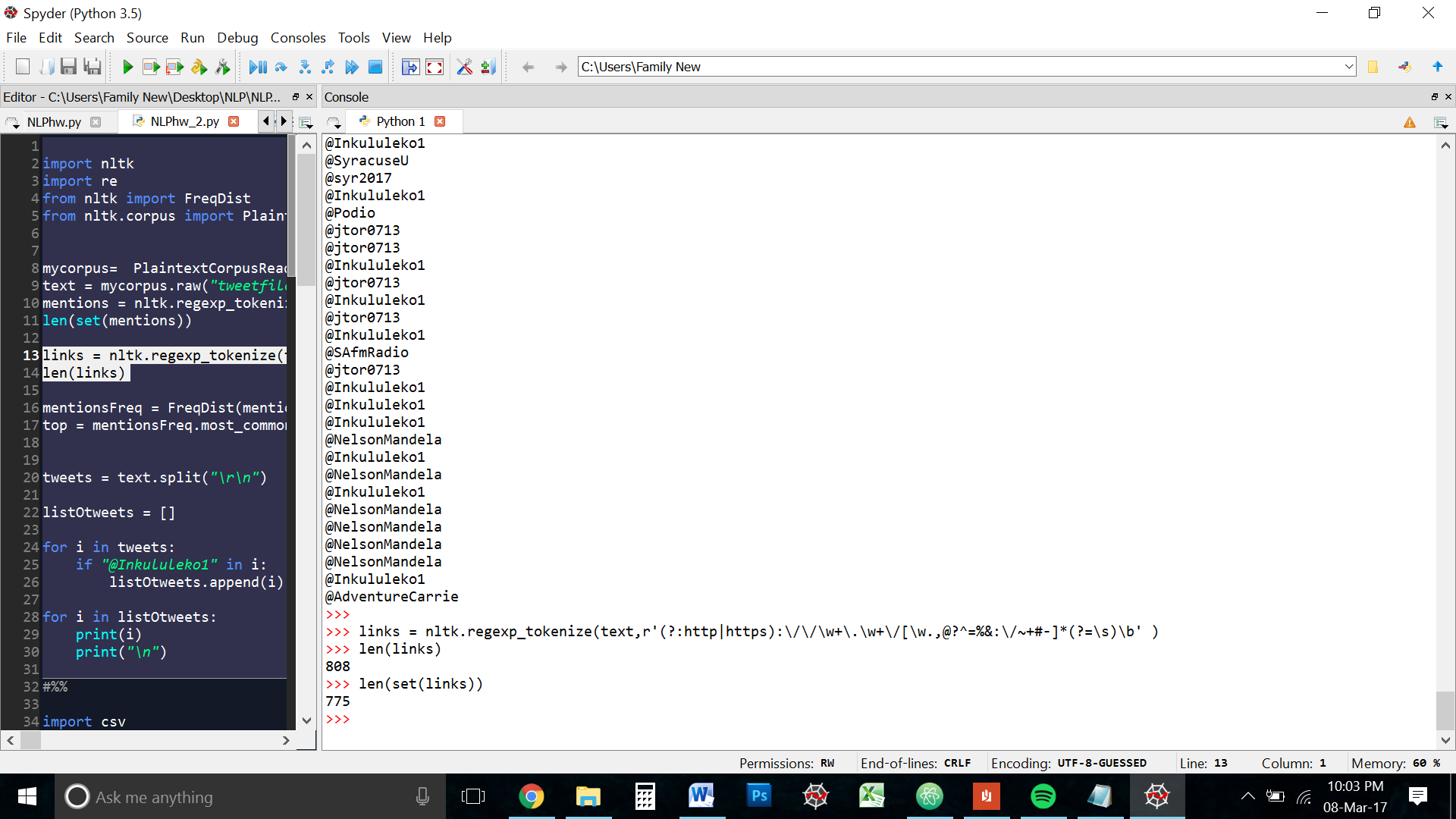
*This matches Zero or more of everything after the forward slash and included special characters.*

***(?=\s)***

*This is a positive lookahead. What this is doing is making sure that is matching everything from the http|https up until a white space character. This ensures we match the whole link and not just part or if.*

NOTE: The expression I wrote only matched full links, it doesn’t match broken or incomplete ones like “http://…” of which there are many in the text.

Since the question only askes for the number of web links, I think for this purpose the NLTK regular expression tokenizer is enough tokenizer with the Regex is enough.

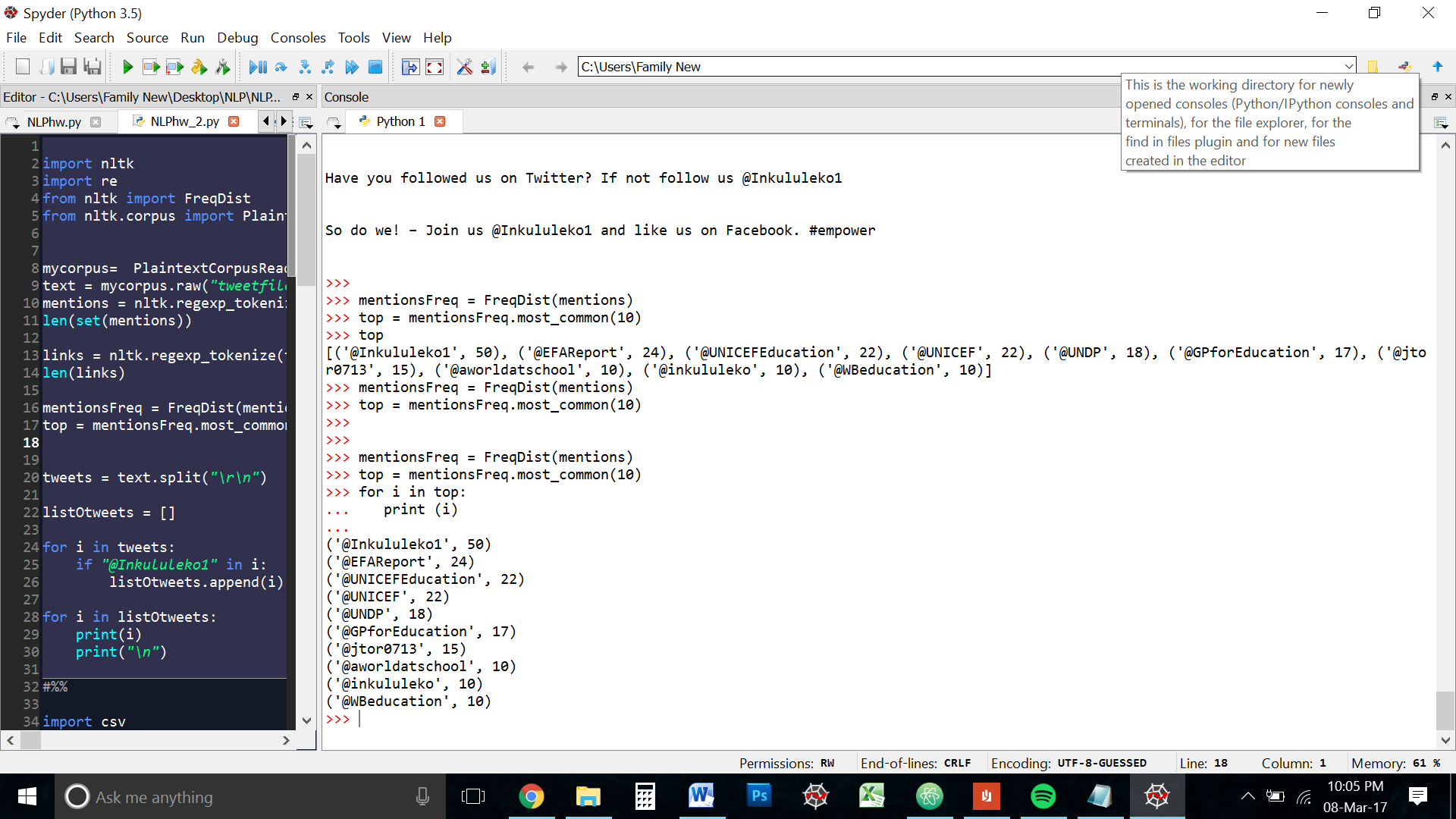


With the above regular expression, there are **808 links** and **775 unique** links

* 1. **Which twitter account was mentioned the most in this data set? How many times?**

Since we need to find the frequency distribution of the accounts mentioned in data, we need the list with repeats, hence we pass the mentions list and not the set.

We then find the most common frequency distribution. I chose the top 5 because I wanted a deeper understanding of the data.

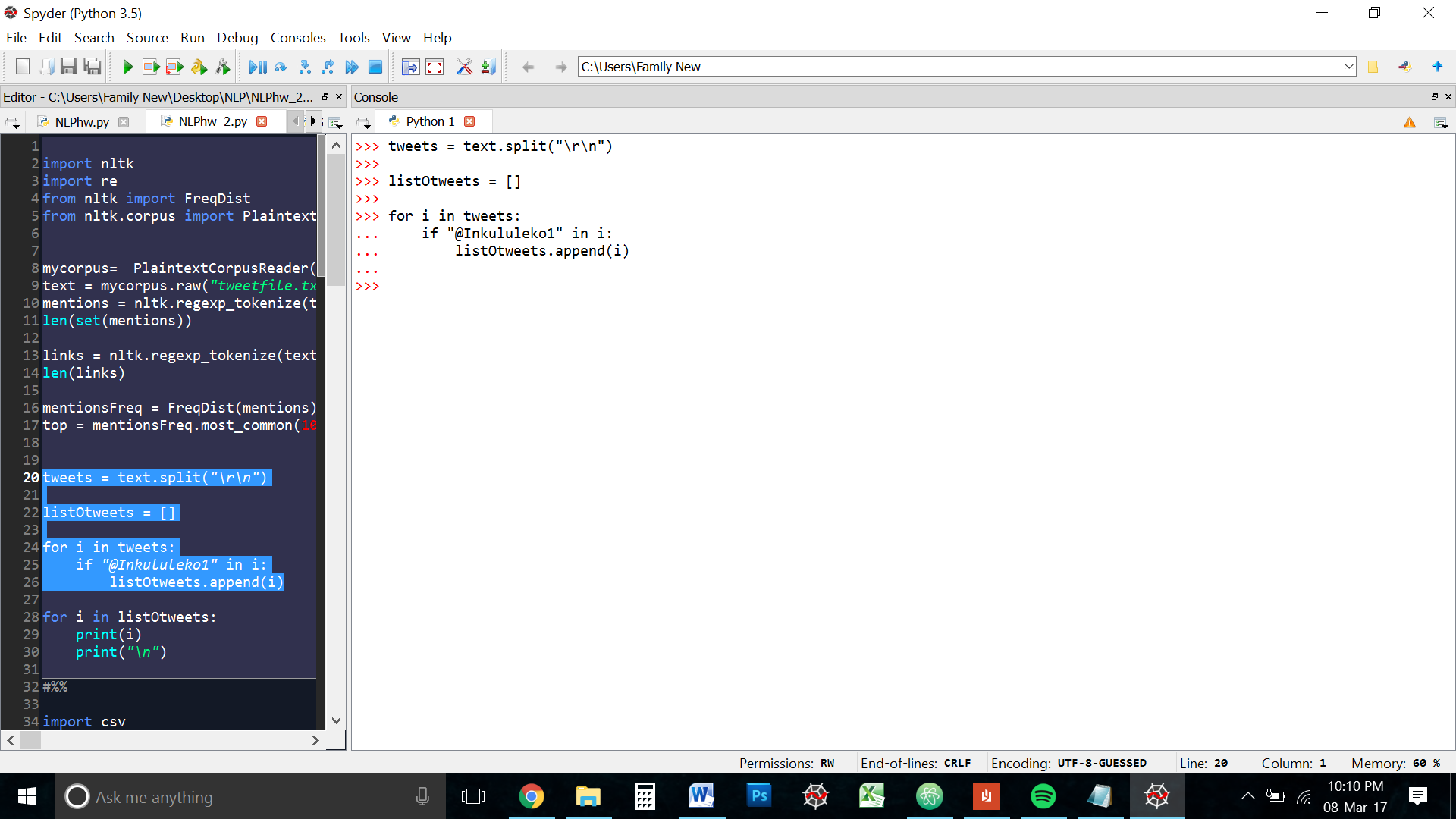


The organization’s own twitter handled is mentioned the most. **50 times** to be exact.

* 1. **For the twitter account that was mentioned the most, find the tweets that mentioned it Do these tweets indicate something about this twitter account? How or why not?**

As mentioned in the beginning, the tweets in the in the file are separated by the characters “\r\n”

I decided to do a simple ‘split’ on the whole text string because this way I will get a clean list of the all the tweets and I can iterate over that list to find what I am looking for, which in this case is the Organizations user handle.



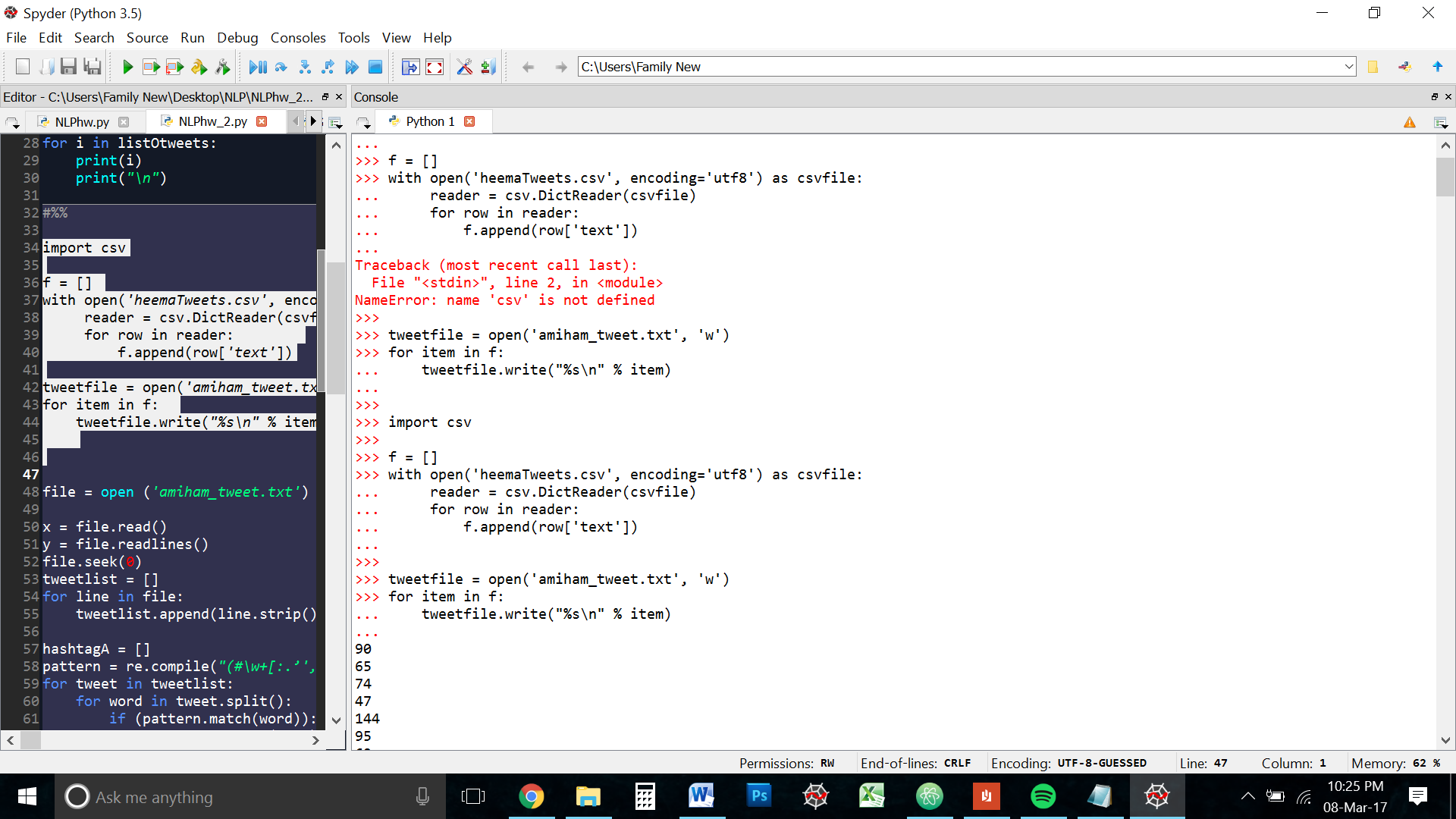
Looking at the list of tweets that contain the username @Inkululeko1, It is apparent that the account has been mentioning itself in form of promotions and self-advertising. This is generally considered bad practice in social media marketing. The goal should be to get the audience talking about you. While there are a few tweets by other users that mention Inkululeko, it is very less compared to the self-promotion the organization is doing.

I also noticed during this analysis that Inkululeko is a very employee friendly organization. In many tweets, it mentions the usernames of employees and interns.

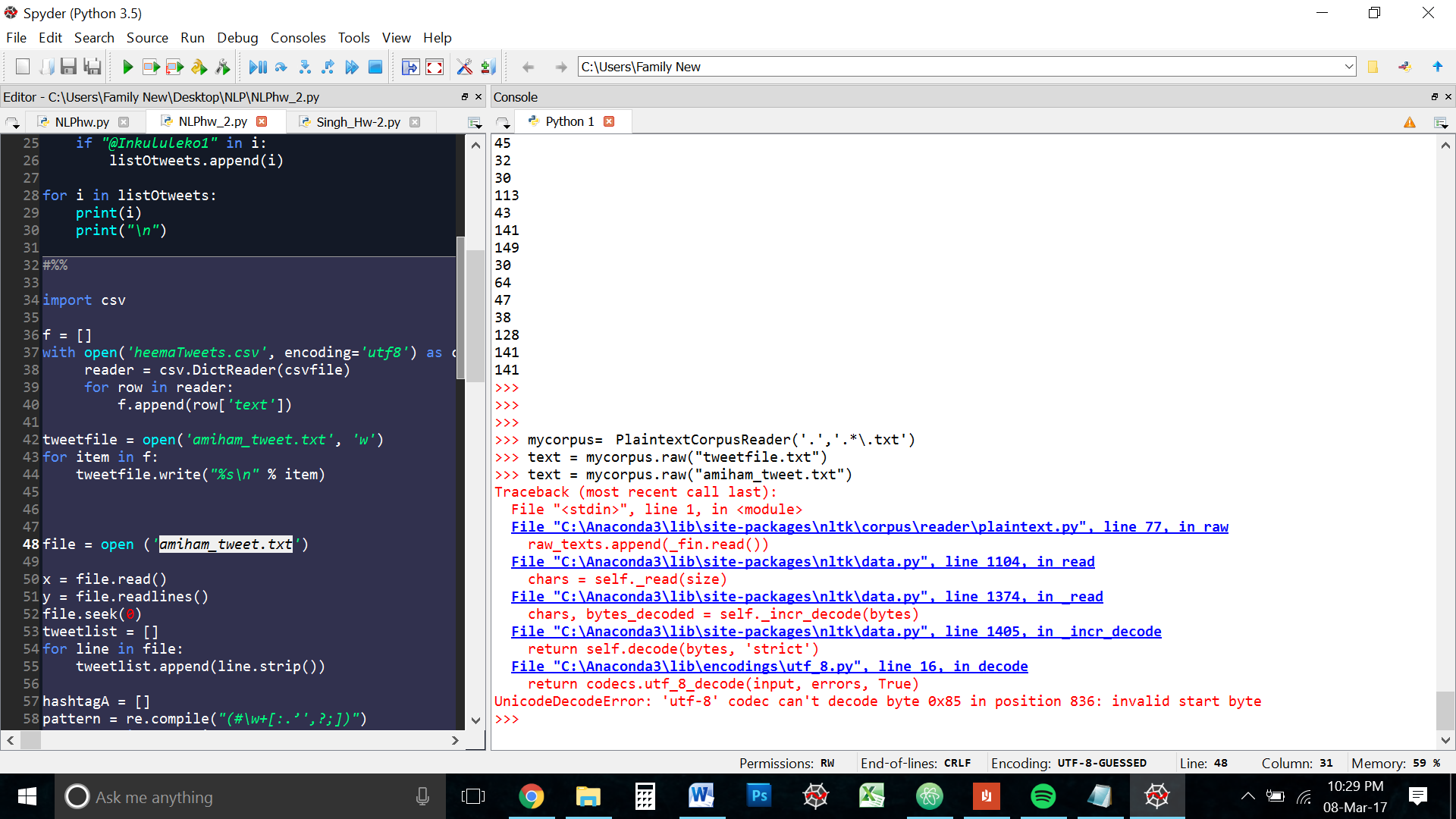
1. **The hashtags that I have used in the past**

I followed the instructions given in the HW file to download my tweets. Going through some of my old tweets was a sweet reminder of how we change as people.

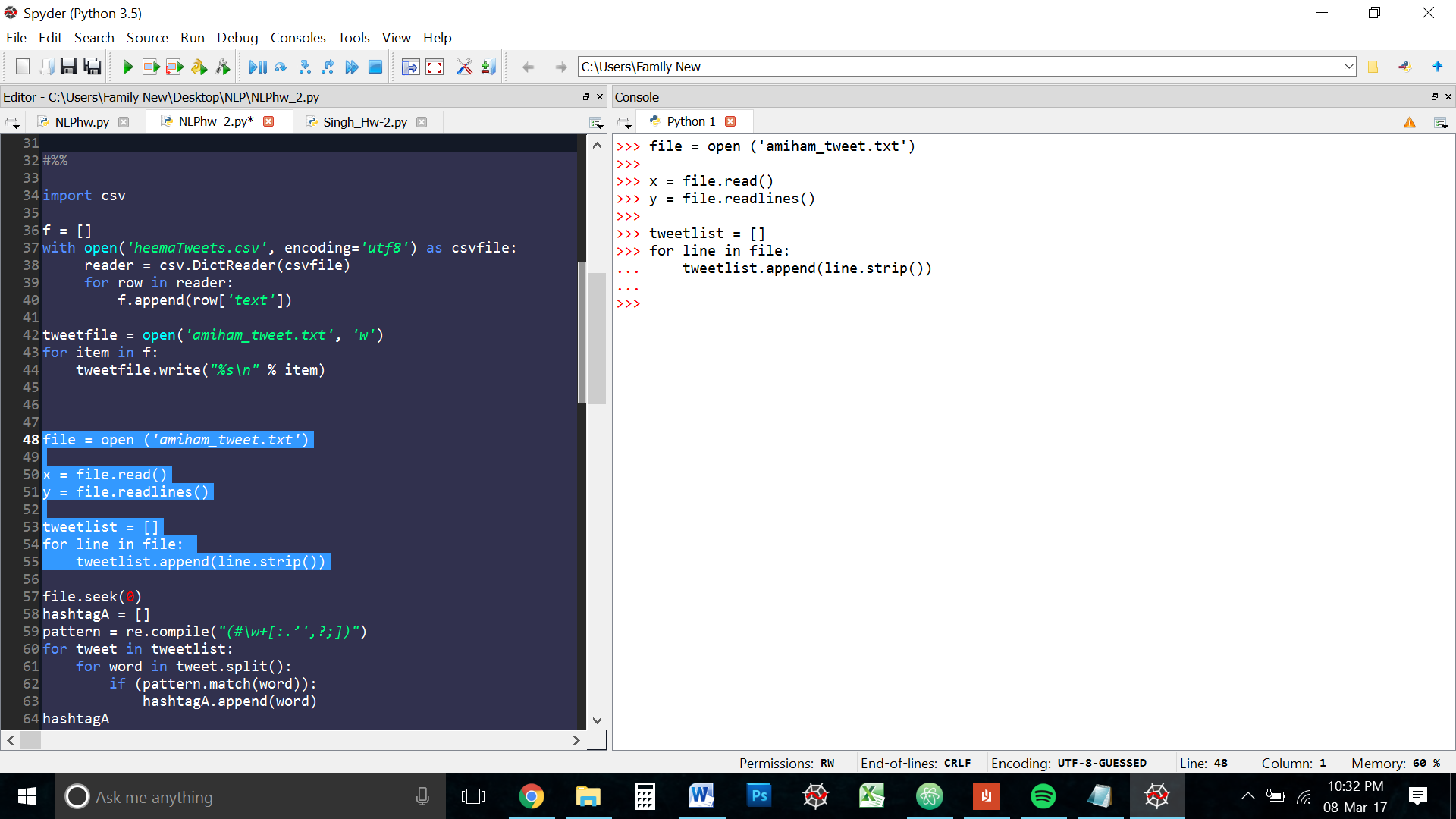
I converted the cvs into a text file.



I tried to tokenize the text file, but I kept encountering an error about the encoding



Even after I tried to research the problem, I was getting nowhere. So I decided to think outside the box for a solution. I read the tweets from the file, line by line and append them to a list.



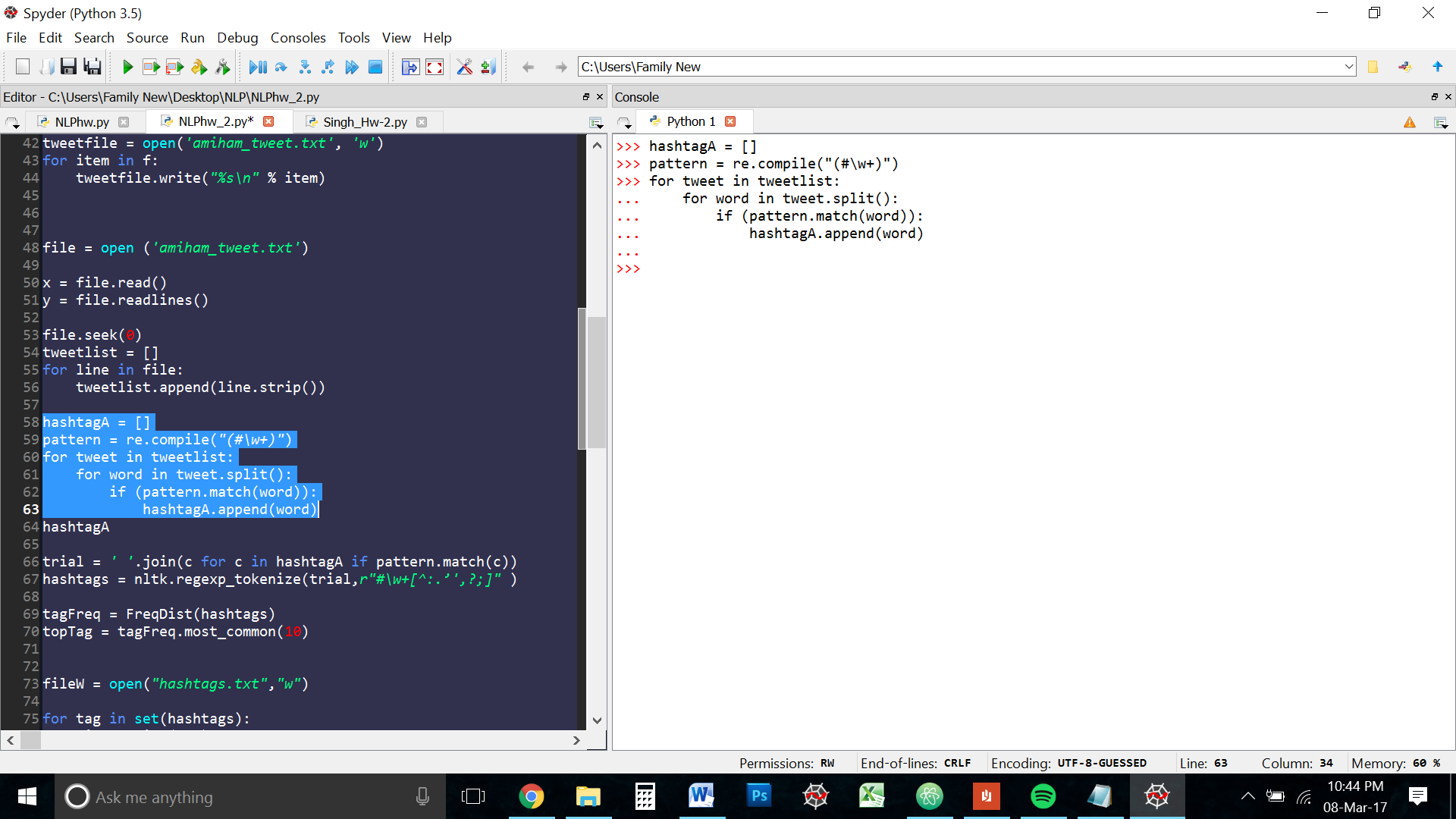
I then wrote a regular expression to match the hashtags

**(#\w+)**

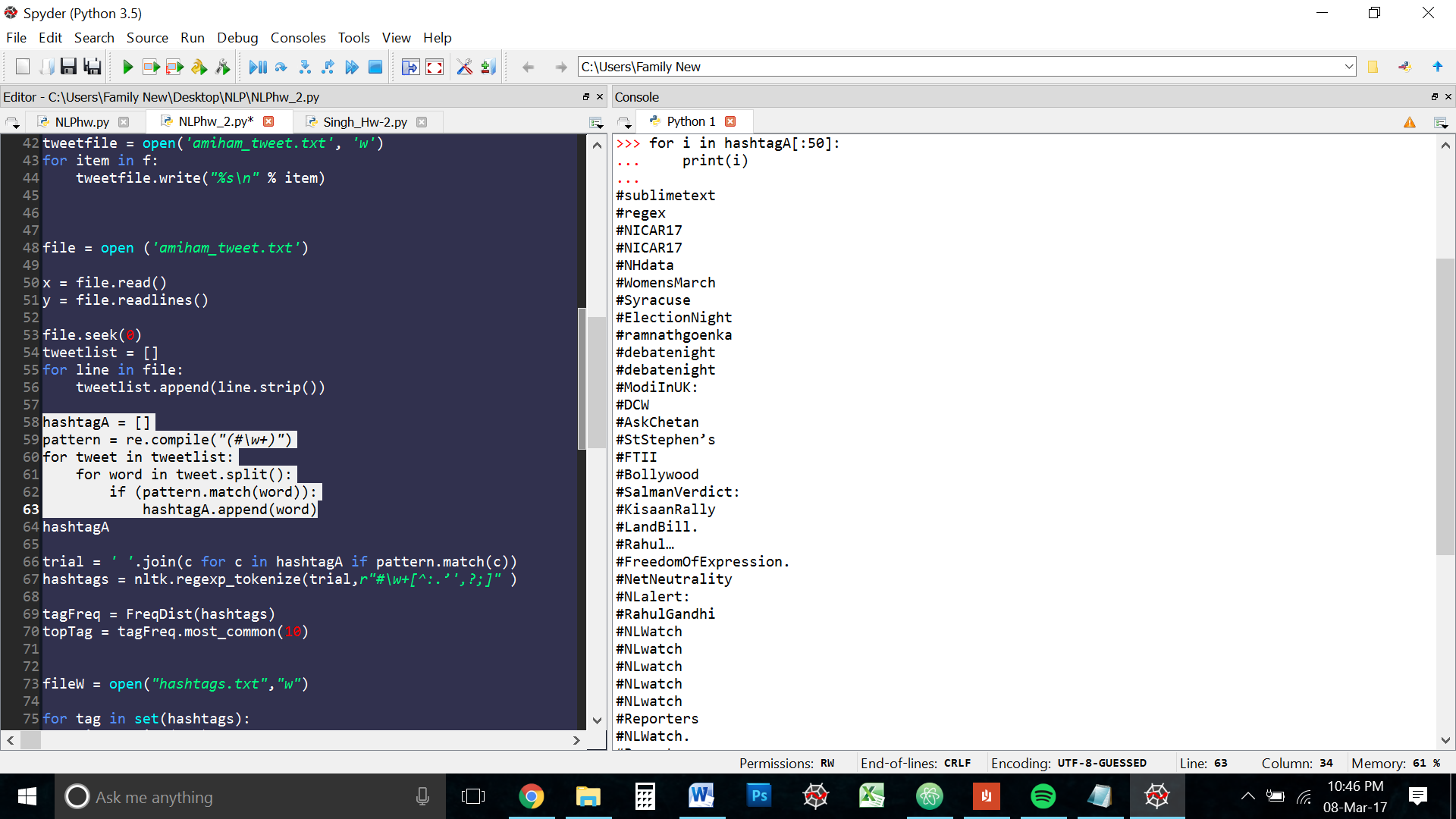
*Below is a breakdown of the final Regex for this question*

*The regular expression is inside a grouping bracket so that the entire hashtag is captured.*

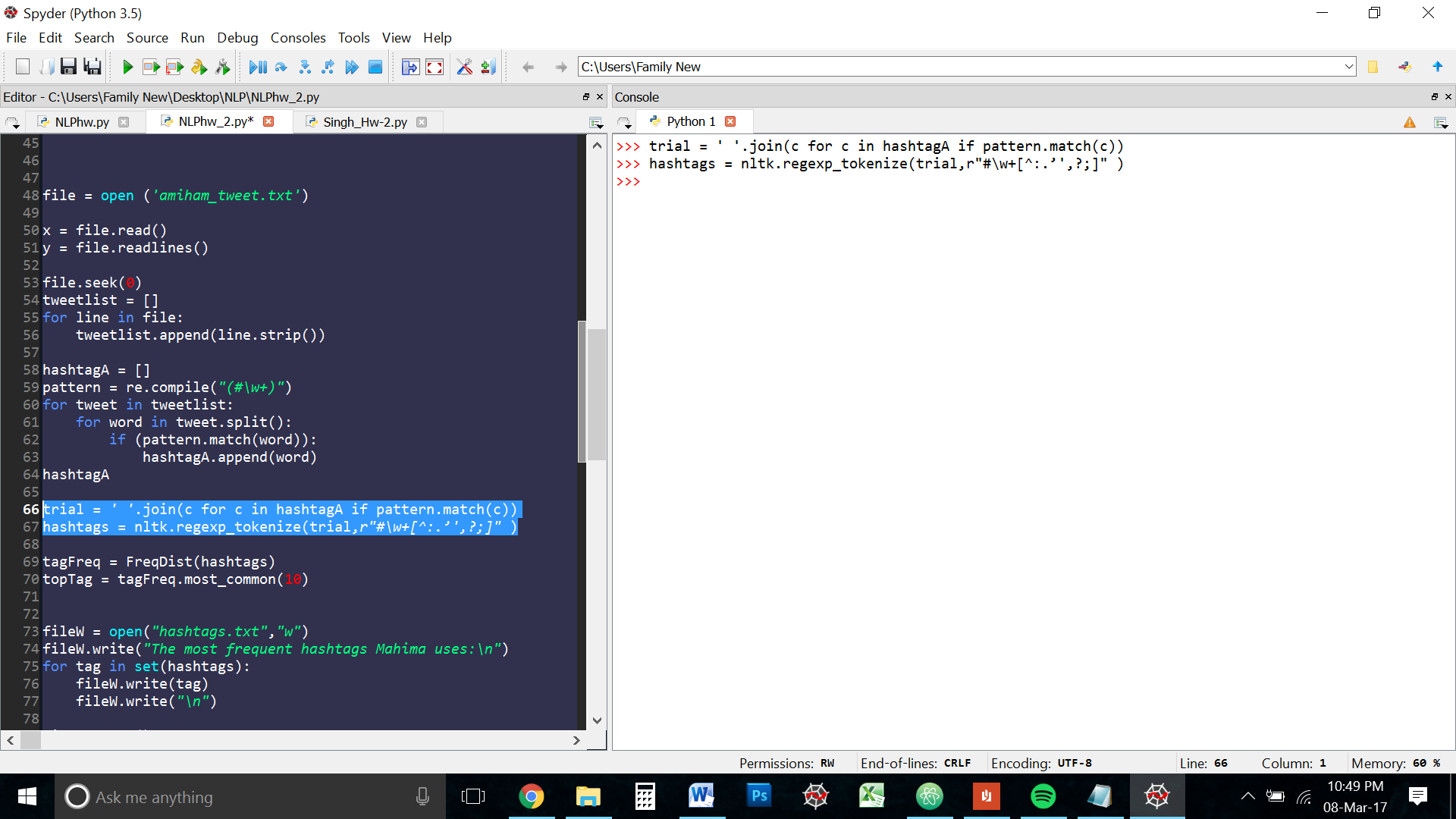
*#\w+ : This matches the hashtag icon and andy word/number or underscore in front of it*



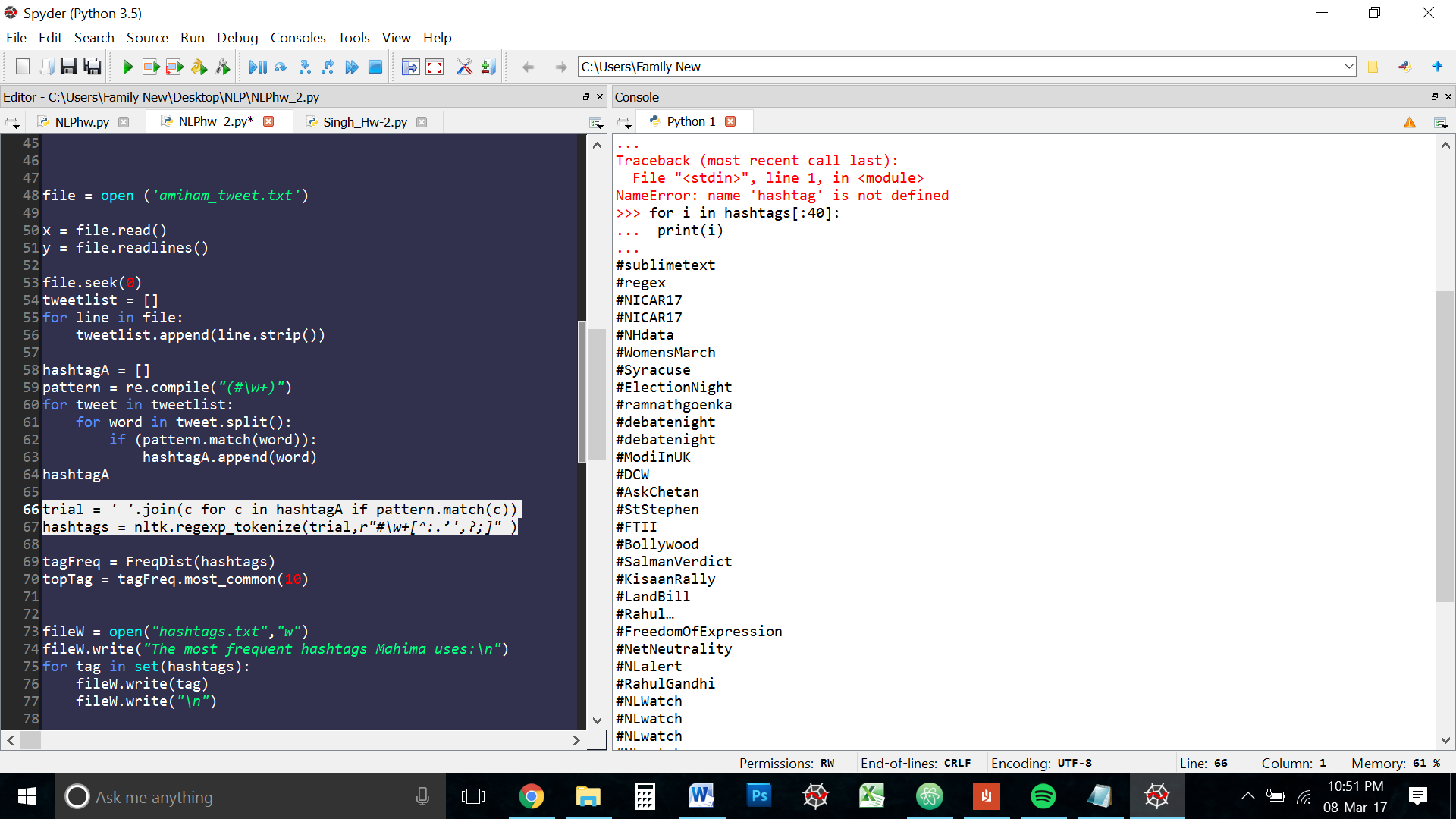
The hashtagA list contains all the hashtags from the file. But there are some unwanted elements at the trailing end of a few. Namely ‘ : . ; ,’ etc



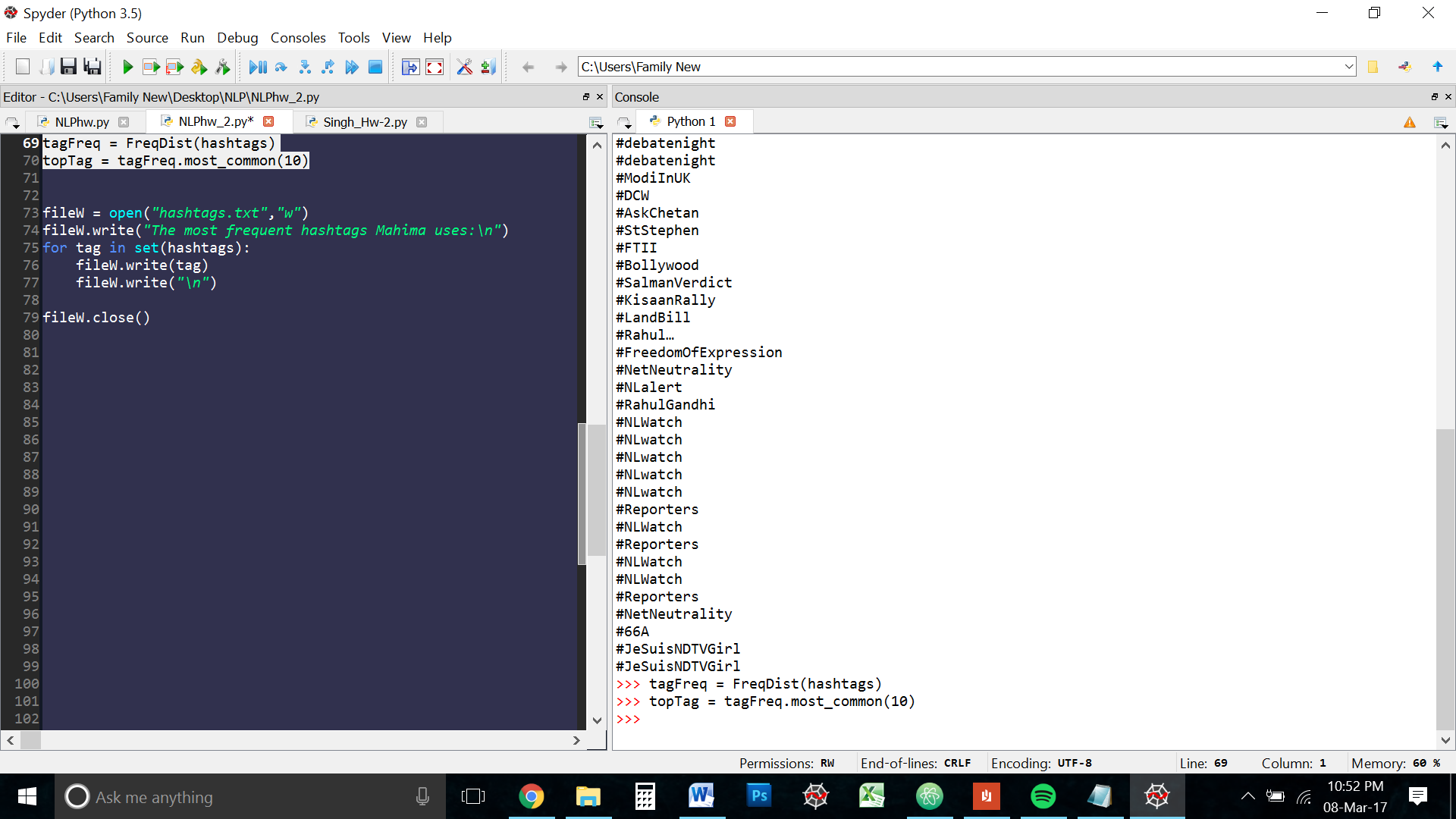
In order to get rid of those unwanted elements, I took the hashtag list converted into one string and the tokenized it with the regexTokenzier.

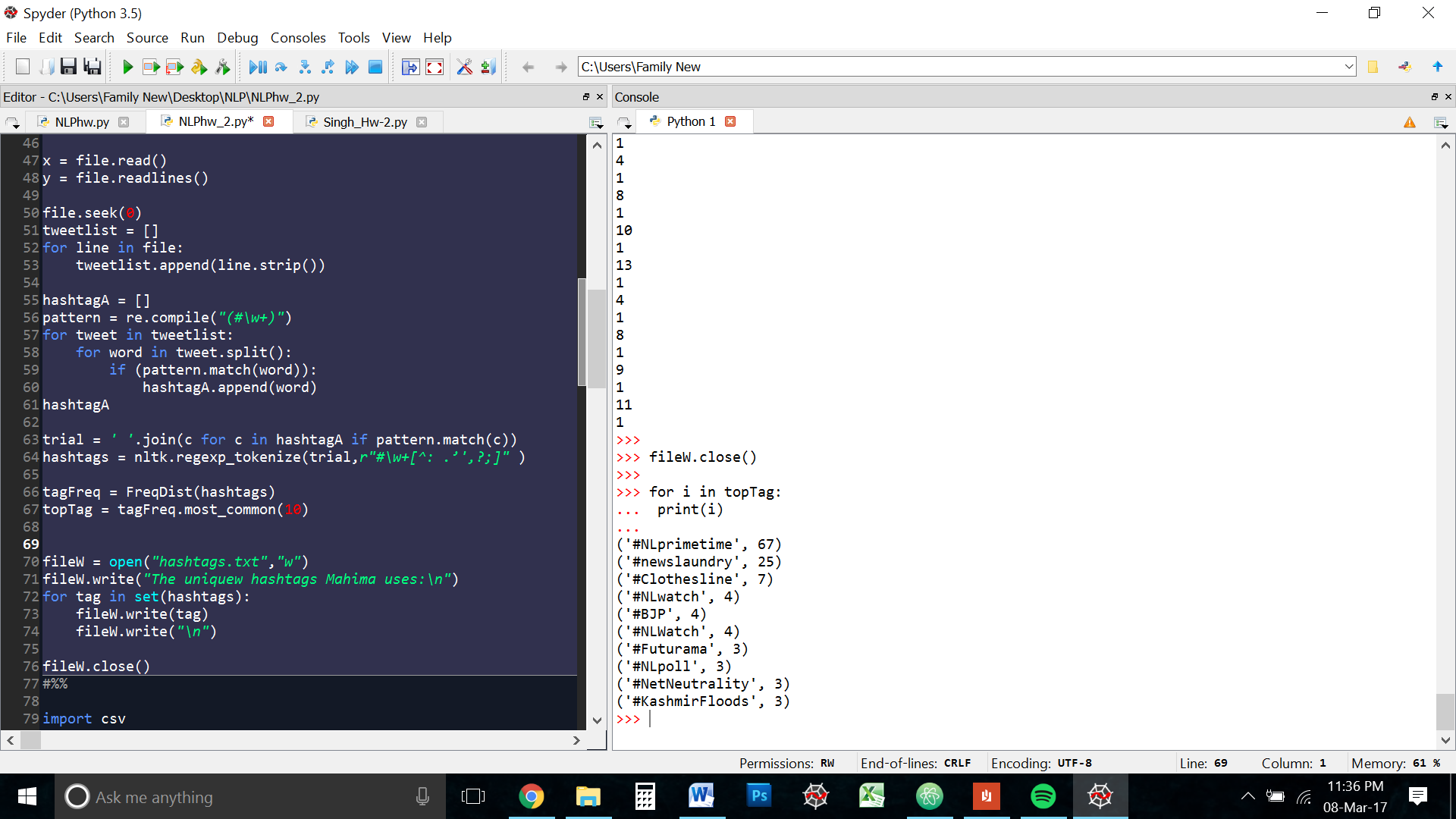


This solved the problem. I now had a tokenized list of all the hashtags.



In order to get the most used hashtags, I ran the tokens through a frequency distribution function and recorded the top 10 hashtags for context.





Then I wrote the unique hashtags to a file



The interpreter is built in such a way that is returns the number of characters written in to a file. Hence the 34 before the for loop

Note: I used the set of the hashtags list here to get the unique hashtags

* 1. **Are there any hashtags that are used more often? If so, what are the three hashtags that appeared more than the rest? Do these hashtags offer any useful information about the use of this Twitter account? Why or why not?**

Since this is my personal account, most of the hashtags were those of my previous employers. Which is understandable, as I used to manage their social media and retweet them a lot.

The three most used hashtags are :

#NLprimetime

#Newslaundry

#Clothesline

Newslaundry is the name of the organisation I used ot work for. NlPrimetime and clotheslines are two segmetns of the website.

Appendix

CODE

import nltk

import re

from nltk import FreqDist

from nltk.corpus import PlaintextCorpusReader

mycorpus= PlaintextCorpusReader('.','.\*\.txt')

text = mycorpus.raw("tweetfile.txt")

mentions = nltk.regexp\_tokenize(text,r'@\w+\b' )

len(set(mentions))

links = nltk.regexp\_tokenize(text,r'(?:http|https):\/\/\w+\.\w+\/[\w.,@?^=%&:\/~+#-]\*(?=\s)\b' )

len(links)

mentionsFreq = FreqDist(mentions)

top = mentionsFreq.most\_common(10)

tweets = text.split("\r\n")

listOtweets = []

for i in tweets:

if "@Inkululeko1" in i:

listOtweets.append(i)

#QUESTION 2

import csv

f = []

with open('heemaTweets.csv', encoding='utf8') as csvfile:

reader = csv.DictReader(csvfile)

for row in reader:

f.append(row['text'])

tweetfile = open('amiham\_tweet.txt', 'w')

for item in f:

tweetfile.write("%s\n" % item)

file = open ('amiham\_tweet.txt')

tweetlist = []

for line in file:

tweetlist.append(line.strip())

hashtagA = []

pattern = re.compile("(#\w+)")

for tweet in tweetlist:

for word in tweet.split():

if (pattern.match(word)):

hashtagA.append(word)

trial = ' '.join(c for c in hashtagA if pattern.match(c))

hashtags = nltk.regexp\_tokenize(trial,r"#\w+[^:.’',?;]" )

tagFreq = FreqDist(hashtags)

topTag = tagFreq.most\_common(10)

fileW = open("hashtags.txt","w")

fileW.write("The uniquew hashtags Mahima uses:\n")

for tag in set(hashtags):

fileW.write(tag)

fileW.write("\n")

fileW.close()

end