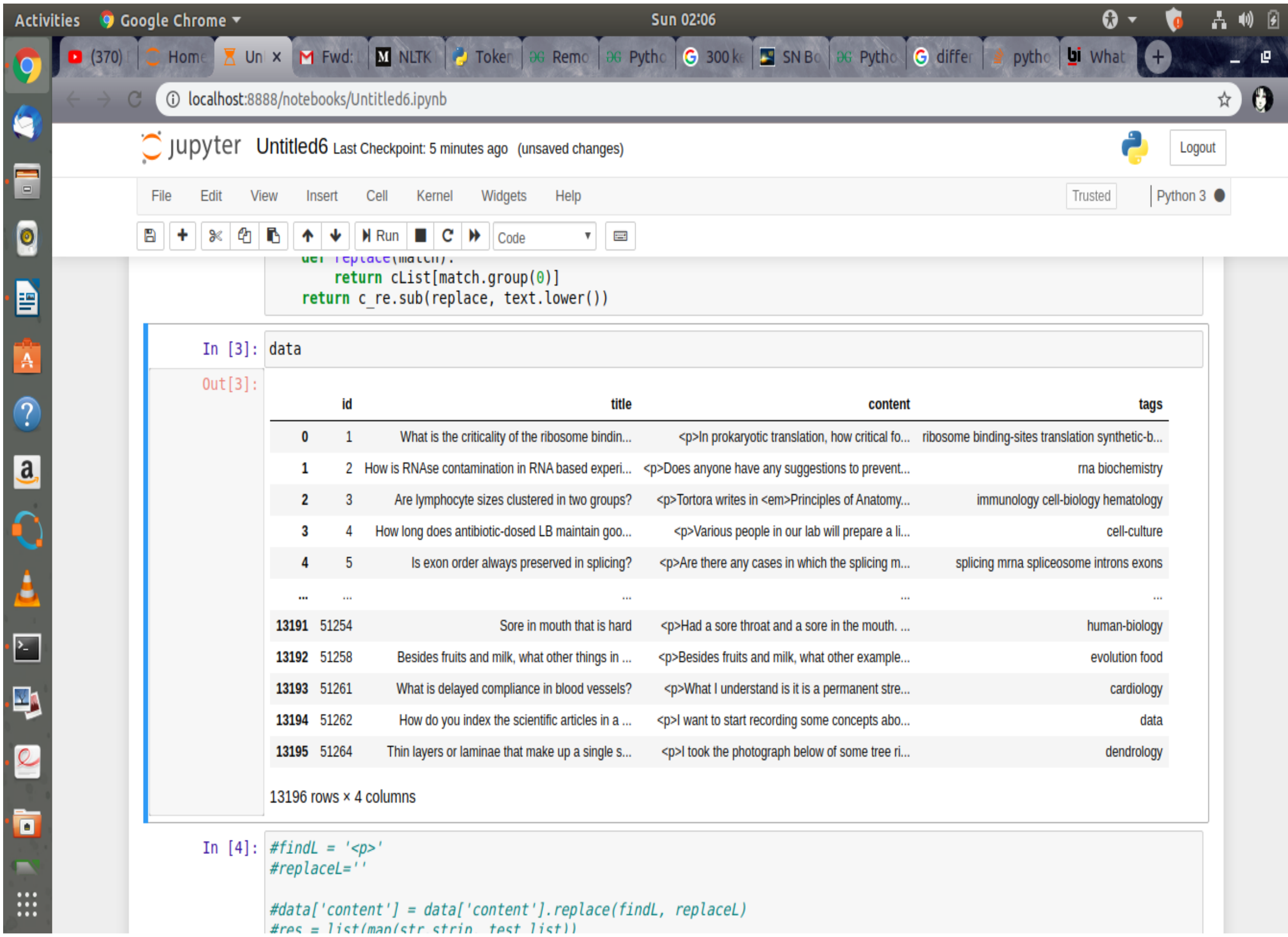
Assignment 1

Step 1: packages which we have used

```
import numpy as np

import os

import html as ihtml

import pandas as pd

from pandas import DataFrame

import csv

import re

import nltk

from urllib.request import urlopen

from bs4 import BeautifulSoup

from nltk.tokenize import sent_tokenize, word_tokenize

from nltk.corpus import stopwords

from nltk.tokenize import word_tokenize

from nltk.stem import WordNetLemmatizer

from nltk.stem import PorterStemmer
```

Steps 2: First we have included the file name biology.csv into data using pandas

```
data=pd.read_csv('Downloads/biology.csv',delimiter=',')
```


Activities
Google Chrome
Sun 02:06
localhost:8888/notebooks/Untitled6.ipynb
jupyter Untitled6 Last Checkpoint: 5 minutes ago (unsaved changes)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code

```
    def replace(match):
        return cList[match.group(0)]
    return c_re.sub(replace, text.lower())
```

```
In [3]: data
Out[3]:
```

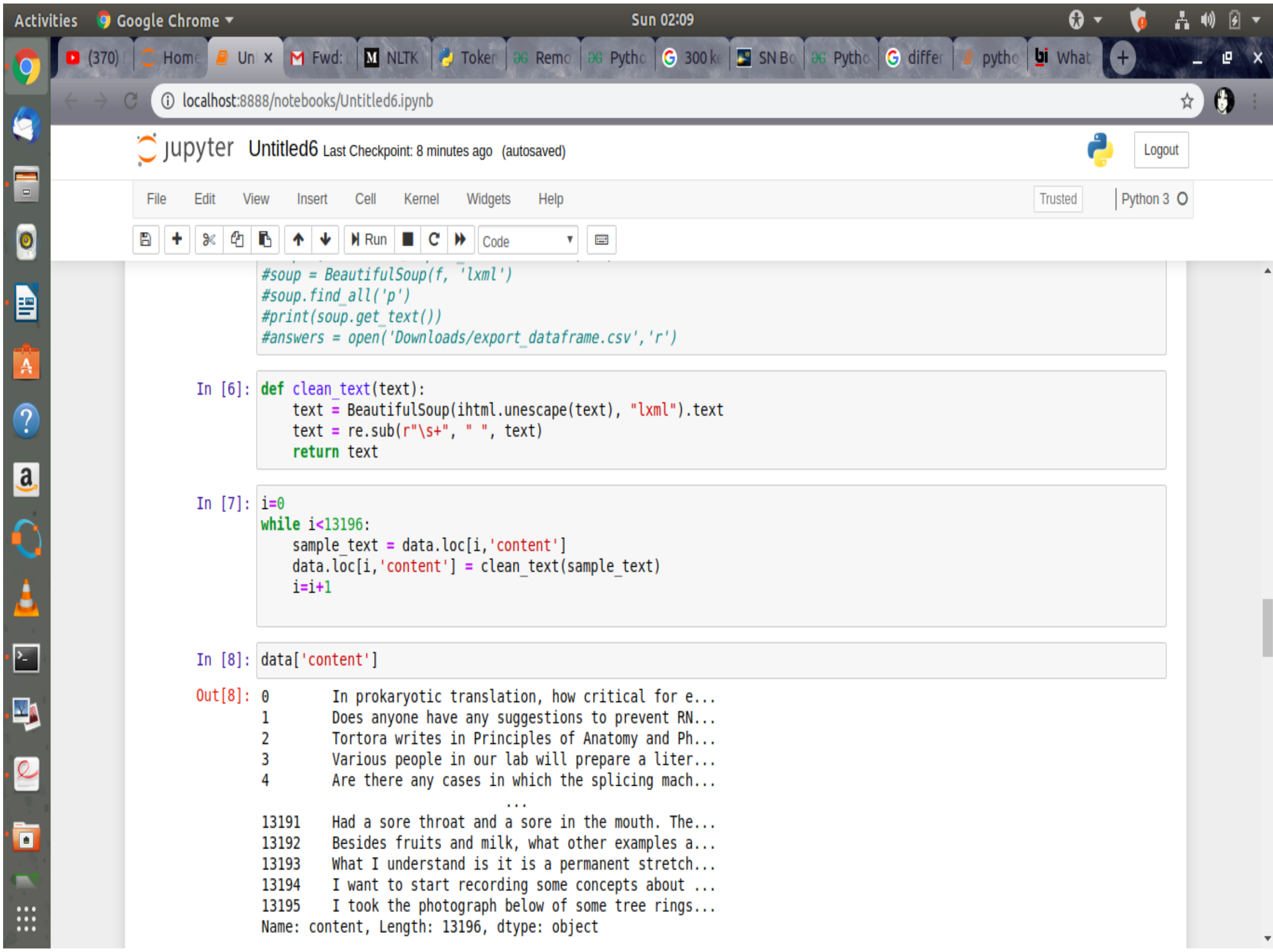
	id	title	content	tags
0	1	What is the criticality of the ribosome bindin...	<p>In prokaryotic translation, how critical fo...	ribosome binding-sites translation synthetic-b...
1	2	How is RNAse contamination in RNA based experi...	<p>Does anyone have any suggestions to prevent...	rna biochemistry
2	3	Are lymphocyte sizes clustered in two groups?	<p>Tortora writes in <em>Principles of Anatomy...	immunology cell-biology hematology
3	4	How long does antibiotic-dosed LB maintain goo...	<p>Various people in our lab will prepare a li...	cell-culture
4	5	Is exon order always preserved in splicing?	<p>Are there any cases in which the splicing m...	splicing mrna spliceosome introns exons
...	...	...	...	...
13191	51254	Sore in mouth that is hard	<p>Had a sore throat and a sore in the mouth. ...	human-biology
13192	51258	Besides fruits and milk, what other things in ...	<p>Besides fruits and milk, what other example...	evolution food
13193	51261	What is delayed compliance in blood vessels?	<p>What I understand is it is a permanent stre...	cardiology
13194	51262	How do you index the scientific articles in a ...	<p>I want to start recording some concepts abo...	data
13195	51264	Thin layers or laminae that make up a single s...	<p>I took the photograph below of some tree ri...	dendrology

13196 rows × 4 columns

```
In [4]: #findL = '<p>'
        #replaceL=''

        #data['content'] = data['content'].replace(findL, replaceL)
        #res = list(map(str.strip, test_list))
```

Step3: In this step we used BeautifulSoup using to remove tags and to get text from tags using function and additionally I have used a while loop to take one block of row at time and get text out of it.

```
sample_text = data.loc[i,'content']

def clean_text(text):

    text = BeautifulSoup(ihtml.unescape(text), "lxml").text

    text = re.sub(r"\s+", " ", text)

    return text
```



```
#soup = BeautifulSoup(f, 'lxml')
#soup.find_all('p')
#print(soup.get_text())
#answers = open('Downloads/export_dataframe.csv','r')
```

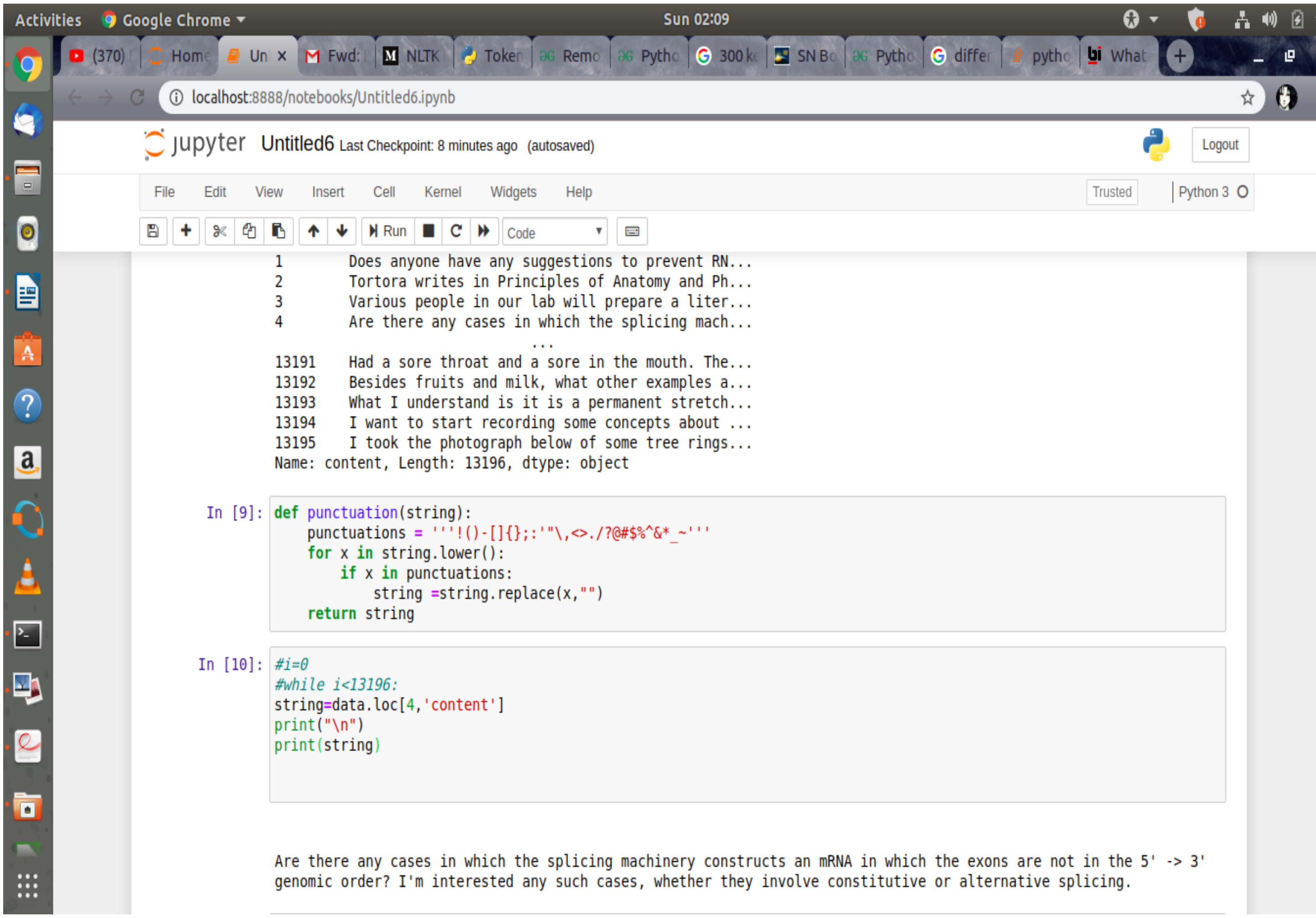
In [6]:
```
def clean_text(text):
    text = BeautifulSoup(ihtml.unescape(text), "lxml").text
    text = re.sub(r"\s+", " ", text)
    return text
```

In [7]:
```
i=0
while i<13196:
    sample_text = data.loc[i,'content']
    data.loc[i,'content'] = clean_text(sample_text)
    i=i+1
```

In [8]:
```
data['content']
```

Out[8]:
```
0        In prokaryotic translation, how critical for e...
1        Does anyone have any suggestions to prevent RN...
2        Tortora writes in Principles of Anatomy and Ph...
3        Various people in our lab will prepare a liter...
4        Are there any cases in which the splicing mach...
                               ...
13191    Had a sore throat and a sore in the mouth. The...
13192    Besides fruits and milk, what other examples a...
13193    What I understand is it is a permanent stretch...
13194    I want to start recording some concepts about ...
13195    I took the photograph below of some tree rings...
Name: content, Length: 13196, dtype: object
```

Step 4: After that I have taken one string and perform different operations


Activities
Google Chrome
Sun 02:09
localhost:8888/notebooks/Untitled6.ipynb
jupyter Untitled6 Last Checkpoint: 8 minutes ago (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Widgets
Help
Trusted
Python 3
Run
Code

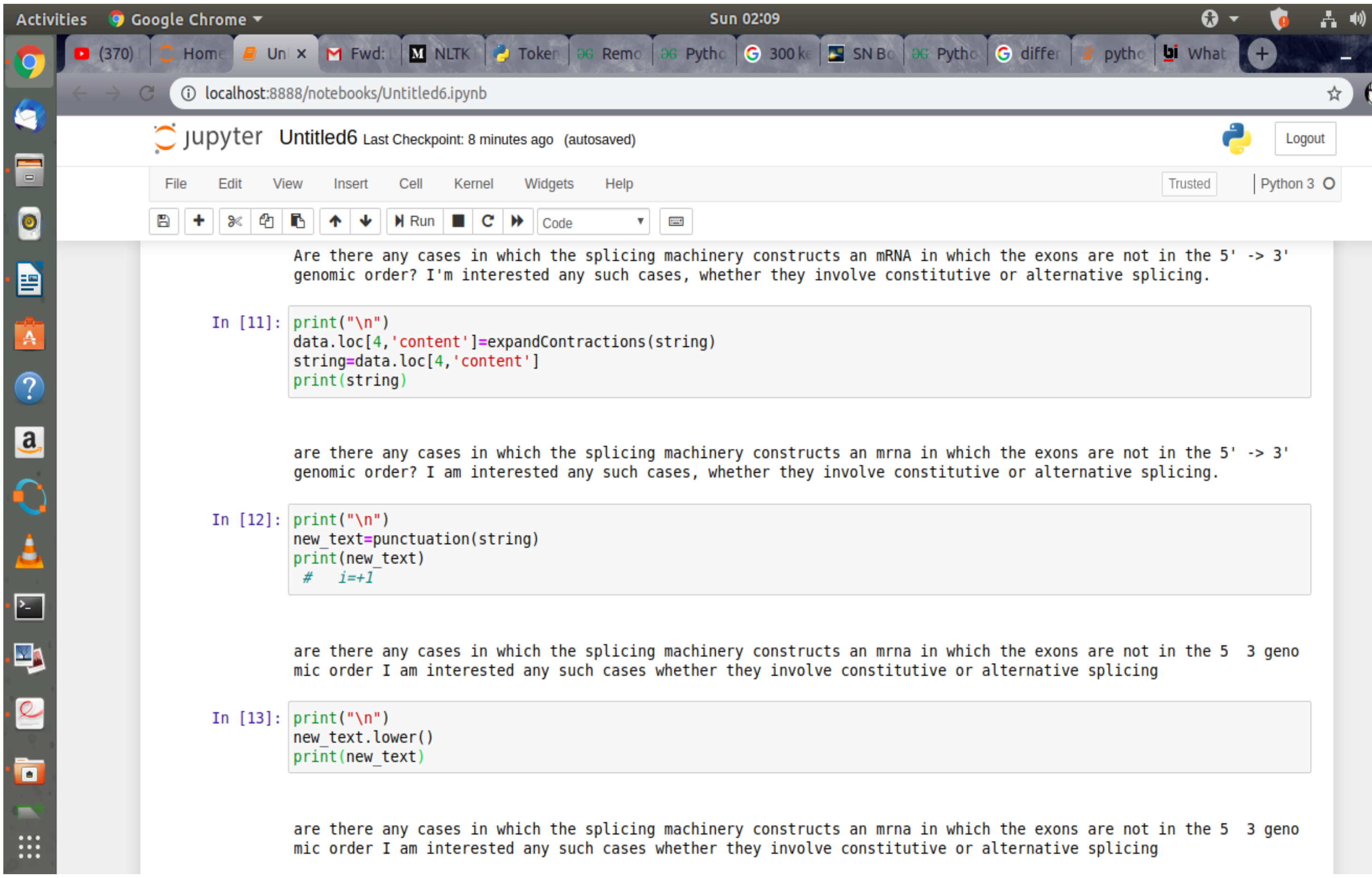
```
1        Does anyone have any suggestions to prevent RN...
2        Tortora writes in Principles of Anatomy and Ph...
3        Various people in our lab will prepare a liter...
4        Are there any cases in which the splicing mach...
                                ...
13191    Had a sore throat and a sore in the mouth. The...
13192    Besides fruits and milk, what other examples a...
13193    What I understand is it is a permanent stretch...
13194    I want to start recording some concepts about ...
13195    I took the photograph below of some tree rings...
Name: content, Length: 13196, dtype: object
```

```
In [9]: def punctuation(string):
            punctuations = '''!()-[]{};:'"\,<>./?@#$%^&*_~'''
            for x in string.lower():
                if x in punctuations:
                    string =string.replace(x,"")
            return string
```

```
In [10]: #i=0
         #while i<13196:
         string=data.loc[4,'content']
         print("\n")
         print(string)
```

```
Are there any cases in which the splicing machinery constructs an mRNA in which the exons are not in the 5' -> 3'
genomic order? I'm interested any such cases, whether they involve constitutive or alternative splicing.
```

string=data.loc[4,'content'] : it used to access local data



```
Are there any cases in which the splicing machinery constructs an mRNA in which the exons are not in the 5' -> 3'
genomic order? I'm interested any such cases, whether they involve constitutive or alternative splicing.
```

In [11]:

```
print("\n")
data.loc[4,'content']=expandContractions(string)
string=data.loc[4,'content']
print(string)
```

```
are there any cases in which the splicing machinery constructs an mrna in which the exons are not in the 5' -> 3'
genomic order? I am interested any such cases, whether they involve constitutive or alternative splicing.
```

In [12]:

```
print("\n")
new_text=punctuation(string)
print(new_text)
 #   i=+1
```

```
are there any cases in which the splicing machinery constructs an mrna in which the exons are not in the 5  3 geno
mic order I am interested any such cases whether they involve constitutive or alternative splicing
```

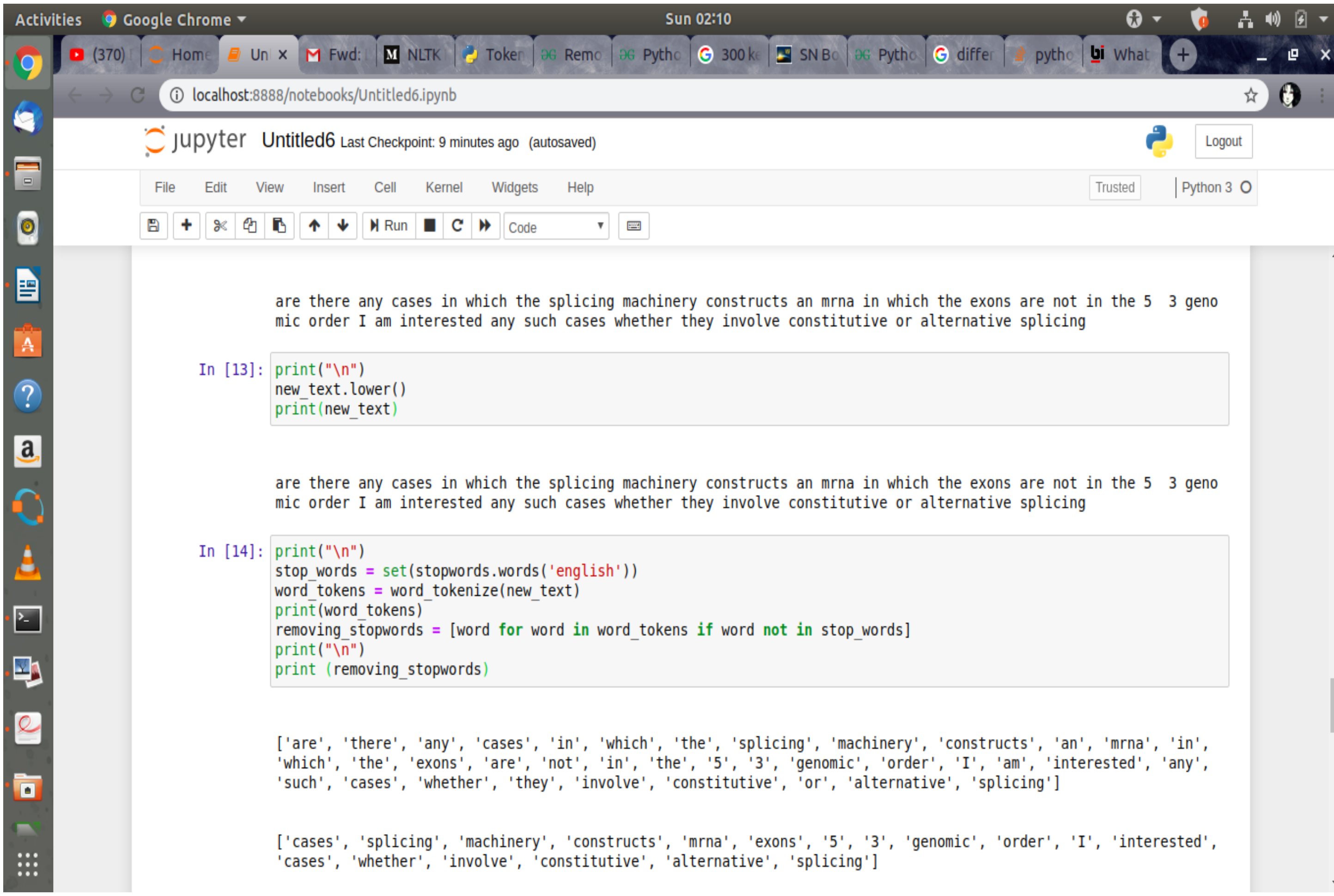
In [13]:

```
print("\n")
new_text.lower()
print(new_text)
```

```
are there any cases in which the splicing machinery constructs an mrna in which the exons are not in the 5  3 geno
mic order I am interested any such cases whether they involve constitutive or alternative splicing
```

data.loc[4,'content']=expandContractions(string): it is used for Exapand words like could’ve to could have

string=data.loc[4,'content']


Activities
Google Chrome
Sun 02:10
localhost:8888/notebooks/Untitled6.ipynb
jupyter Untitled6 Last Checkpoint: 9 minutes ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
are there any cases in which the splicing machinery constructs an mrna in which the exons are not in the 5  3 geno
mic order I am interested any such cases whether they involve constitutive or alternative splicing
```

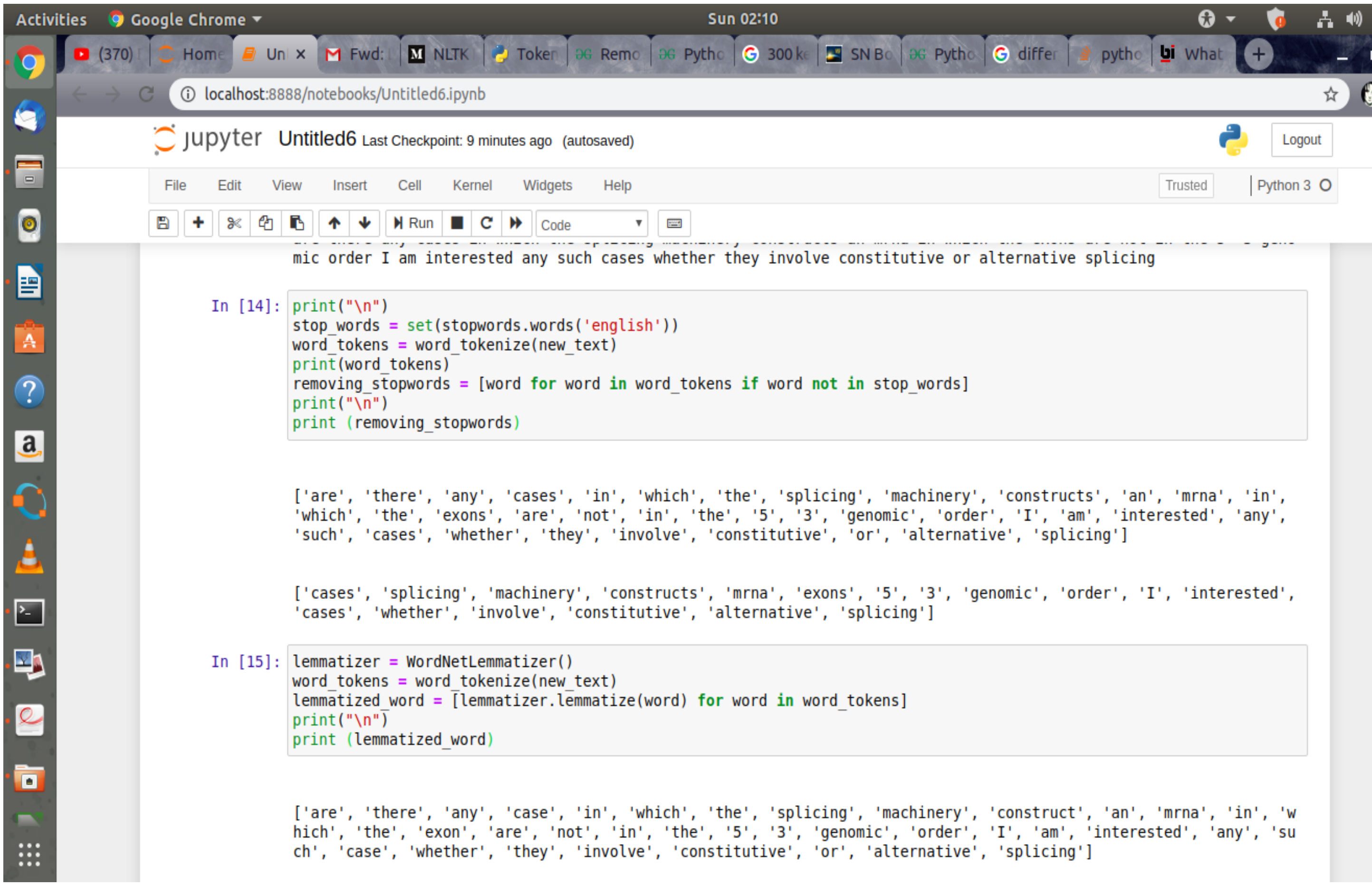
```
In [13]: print("\n")
         new_text.lower()
         print(new_text)
```

```
are there any cases in which the splicing machinery constructs an mrna in which the exons are not in the 5  3 geno
mic order I am interested any such cases whether they involve constitutive or alternative splicing
```

```
In [14]: print("\n")
         stop_words = set(stopwords.words('english'))
         word_tokens = word_tokenize(new_text)
         print(word_tokens)
         removing_stopwords = [word for word in word_tokens if word not in stop_words]
         print("\n")
         print (removing_stopwords)
```

```
['are', 'there', 'any', 'cases', 'in', 'which', 'the', 'splicing', 'machinery', 'constructs', 'an', 'mrna', 'in',
'which', 'the', 'exons', 'are', 'not', 'in', 'the', '5', '3', 'genomic', 'order', 'I', 'am', 'interested', 'any',
'such', 'cases', 'whether', 'they', 'involve', 'constitutive', 'or', 'alternative', 'splicing']


['cases', 'splicing', 'machinery', 'constructs', 'mrna', 'exons', '5', '3', 'genomic', 'order', 'I', 'interested',
'cases', 'whether', 'involve', 'constitutive', 'alternative', 'splicing']
```

In this we tokenize our string of data and removed stop words


Activities
Google Chrome
Sun 02:10
localhost:8888/notebooks/Untitled6.ipynb
jupyter Untitled6 Last Checkpoint: 9 minutes ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
mic order I am interested any such cases whether they involve constitutive or alternative splicing
```

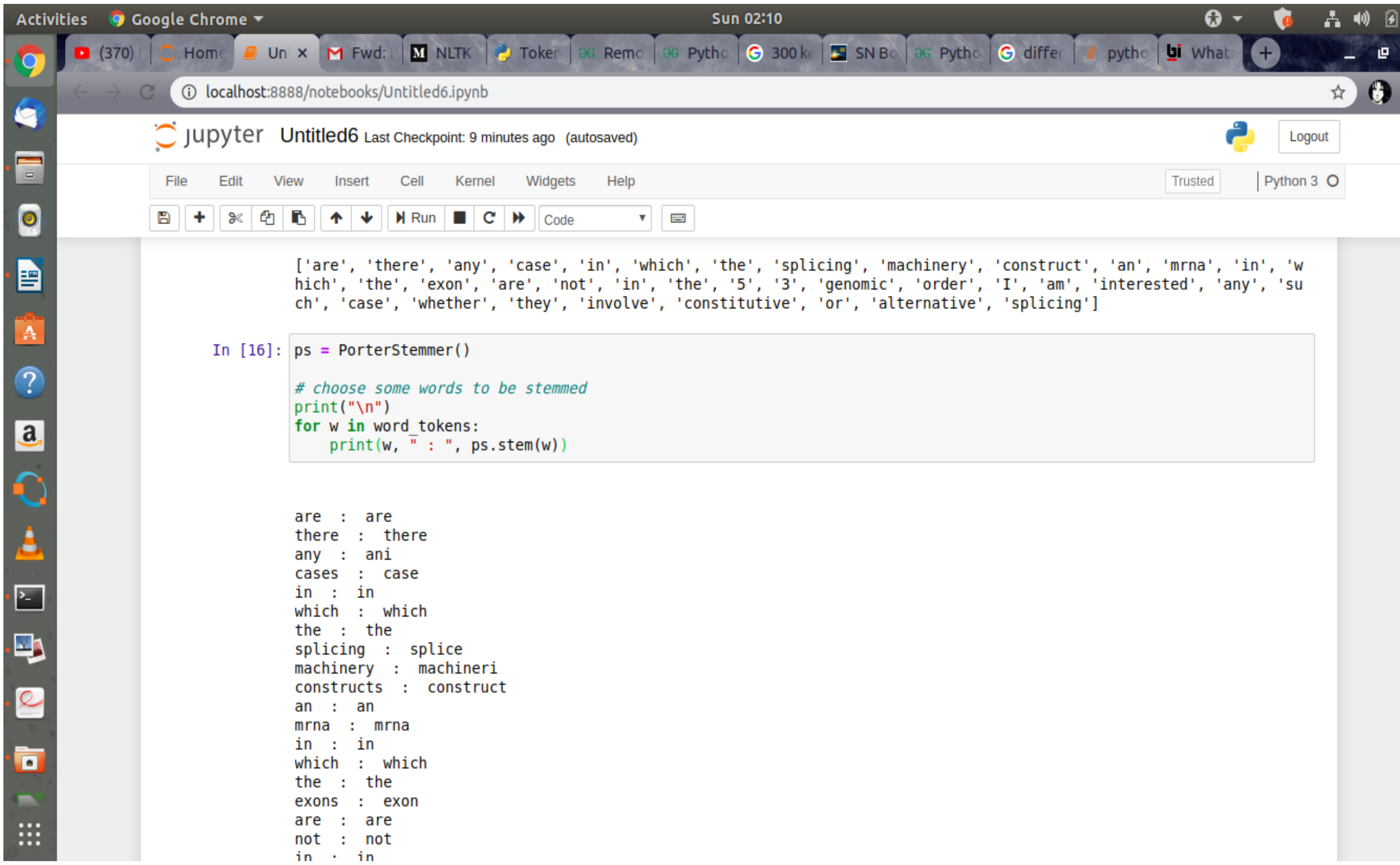
```
In [14]: print("\n")
         stop_words = set(stopwords.words('english'))
         word_tokens = word_tokenize(new_text)
         print(word_tokens)
         removing_stopwords = [word for word in word_tokens if word not in stop_words]
         print("\n")
         print (removing_stopwords)
```

```


['are', 'there', 'any', 'cases', 'in', 'which', 'the', 'splicing', 'machinery', 'constructs', 'an', 'mrna', 'in',
'which', 'the', 'exons', 'are', 'not', 'in', 'the', '5', '3', 'genomic', 'order', 'I', 'am', 'interested', 'any',
'such', 'cases', 'whether', 'they', 'involve', 'constitutive', 'or', 'alternative', 'splicing']


['cases', 'splicing', 'machinery', 'constructs', 'mrna', 'exons', '5', '3', 'genomic', 'order', 'I', 'interested',
'cases', 'whether', 'involve', 'constitutive', 'alternative', 'splicing']
```

```
In [15]: lemmatizer = WordNetLemmatizer()
         word_tokens = word_tokenize(new_text)
         lemmatized_word = [lemmatizer.lemmatize(word) for word in word_tokens]
         print("\n")
         print (lemmatized_word)
```

```


['are', 'there', 'any', 'case', 'in', 'which', 'the', 'splicing', 'machinery', 'construct', 'an', 'mrna', 'in', 'w
hich', 'the', 'exon', 'are', 'not', 'in', 'the', '5', '3', 'genomic', 'order', 'I', 'am', 'interested', 'any', 'su
ch', 'case', 'whether', 'they', 'involve', 'constitutive', 'or', 'alternative', 'splicing']
```

In this we used nltk library to lemmatization


Activities
Google Chrome
Sun 02:10
localhost:8888/notebooks/Untitled6.ipynb
jupyter Untitled6 Last Checkpoint: 9 minutes ago (autosaved)
Logout
File Edit View Insert Cell Kernel Widgets Help
Trusted
Python 3
Run
Code

```
['are', 'there', 'any', 'case', 'in', 'which', 'the', 'splicing', 'machinery', 'construct', 'an', 'mrna', 'in', 'w
hich', 'the', 'exon', 'are', 'not', 'in', 'the', '5', '3', 'genomic', 'order', 'I', 'am', 'interested', 'any', 'su
ch', 'case', 'whether', 'they', 'involve', 'constitutive', 'or', 'alternative', 'splicing']
```

```
In [16]: ps = PorterStemmer()

         # choose some words to be stemmed
         print("\n")
         for w in word_tokens:
             print(w, " : ", ps.stem(w))
```

```


are  :  are
there  :  there
any  :  ani
cases  :  case
in  :  in
which  :  which
the  :  the
splicing  :  splice
machinery  :  machineri
constructs  :  construct
an  :  an
mrna  :  mrna
in  :  in
which  :  which
the  :  the
exons  :  exon
are  :  are
not  :  not
in  :  in
```

In this we also used nltk library to stemming our string

Difference between stemming and lemmatization is:

Lemmatization: Lemmatization is the process of converting the words of a sentence to its dictionary form for example, given the words amusement, amusing, and amused, the lemma for each and all would be amuse.

Stemming: Stemming is the process of converting the words of a sentence to its non-changing portions. In the example of amusing, amusement, and amused above, the stem would be amus. In stemming after converting word to a non-changing portion it may not have any meaning

But in lemmatization after converting word to a root form it will always have meaning.