**Text Analysis - Week 2 Lab Worksheet**

1. A. Write a function that retrieves the relevant html data (the soup) elements from a given url, strips away the html tags, then join the resulting strings to form a document; it takes in a url and a list of the relevant elements, and it returns a string. Use the get\_text() function to obtain the text content of the relevant elements.

B. Apply the above function to the following urls to retrieve data from **h1** and **p** elements:

flu\_urls = [

"https://en.wikipedia.org/wiki/Influenza",

'https://en.wikipedia.org/wiki/Influenza\_virus',

'https://en.wikipedia.org/wiki/Flu\_season',

'https://en.wikipedia.org/wiki/Influenza\_pandemic',

'https://en.wikipedia.org/wiki/Rapid\_influenza\_diagnostic\_test'

]

covid\_urls = [

"https://en.wikipedia.org/wiki/COVID-19\_pandemic",

'https://en.wikipedia.org/wiki/Misinformation\_related\_to\_the\_COVID-19\_pandemic',

'https://en.wikipedia.org/wiki/Treatment\_and\_management\_of\_COVID-19',

'https://en.wikipedia.org/wiki/COVID-19\_testing',

'https://en.wikipedia.org/wiki/Contact\_tracing'

]

climate\_urls = [

"https://en.wikipedia.org/wiki/Climate\_change",

'https://en.wikipedia.org/wiki/Climate\_change\_in\_the\_Arctic',

'https://en.wikipedia.org/wiki/Effects\_of\_global\_warming',

'https://en.wikipedia.org/wiki/Retreat\_of\_glaciers\_since\_1850',

'https://en.wikipedia.org/wiki/Greenhouse\_gas'

]

C. Store the results in 3 lists of 5 documents each.

1. A. Then join the 3 lists of texts and assign it to a variable all\_docs;
2. Also, create a list all\_labels that contains 5 ‘flu’ labels, 5 ‘covid’ labels and 5 ‘climate’ labels.
3. A. In preparation for a classification task, define a function that creates a corpus of labelled documents, each having a corresponding label. The function should take in a list of documents and a list of labels and return a dataframe with 2 columns: one for the documents text, one for the documents label. The order of rows in the dataframe should be randomised (can use the **sample** function from pandas).
4. Apply this function to the joint lists generated in 2.
5. A. Define a function that checks whether the length of the document is greater than 32000 characters (this is how much it can fit in a csv cell): if it is, it should only retain the first 32000 characters.
6. Apply this function to the list of documents.
7. Write the corpus to a csv file using the pandas function to\_csv (ensure you do not include a column for index).
8. Finally, load the data from the csv file into a data frame to check that the write operation was successful; do some initial investigation of the documents to see if there are any possible issues with the text in the documents.

**PRACTICE (optional)**: apply the above workflow to other urls; you need to identify the relevant elements first to pass to the first function.