1. **Explanation of my approach to the solution:**

**Extract the Text from the list of webpages**

* The first task was to extract the text from the links provided. Here, I had to make sure that I only extracted the Title and the main text of the article, nothing else.
* The part that was tricky for me was to calculate the “class” that contained the main text.
* As it turned out, the structure of all the webpages are not the same and the main text was not in the same class for all the webpages. Eventually I figured it out and found that for some webpages, the main text was in the class named “td-post-content tagdiv-type” and for others, it was in the class named “tdb\_single\_content”
* After that, I wrote a code that extracted the required text from the specified tags and stored it in a .txt file.
* Overall, I was able to extract the text from 97 out of the 100 webpages.

**Performing textual analysis and computing variables**

* After extracting the text, the next step was to compute the 13 variables
* First, I created a Python list named “stop\_words” and stored all the stop words present the text files inside the “StopWords” folder in that list.
* Then I also created two Python set variables, that stored all the positive and negative words respectively.
* Next, I created a Pandas DataFrame, named “result” and created the headers of the 15 columns, as per the Output Structure presented in the assignment. The variables that will be computed later will be stored in this dataframe and it will be exprted into a Excel file.
* Then, I created a “for loop”, where in each loop it did the following for all the extracted text files:
  + It calculated all the 13 variables as per the given formula
  + Then it created a new Pandas DataFrame row where it stored all the calculated variables along with the URLID and the link.
  + Finally, it appended the row with the “result” dataframe that I created earlier
* After the loop is completed, the “result” DataFrame contains the computed variables of all the extracted text files.
* Finally, the result is saved into a excel file named “Output.xlsx”, which contains the final result of the computed variables.

1. **How to run the .py file to generate output**

* First, make sure that the folder (“Submission Rohan Patgiri”) is opened in an IDE, otherwise there might be some errors while running the code.
* Then, run the “txt\_extract.py” python code file.
* This code will create a new folder named “extracted\_text”.
* Then it will store the extracted text from the webpages, for each webpage mentioned in the “Input.txt” file, and store it in a .txt file.
* Next run the “analyze.py” python code file.
* This code will compute all the variables and store it in a excel file, named “Output.xlsx”

(NOTE: There might be errors about files not being found. In that case copy the path of the file by right clicking on the file in the IDE and selecting “Copy Path”)

1. **Dependencies required**

The following Libraries are required to run the .py files without any errors:

* Pandas
* NLTK
* BeautifulSoup
* Selenium
* Textblob
* re
* os

Apart from the libraries, the following Folders and files will be needed to run the Python code files:

* Input.xlsx: This file contains the links to all the webpages from which text are to be extracted. Without this file, the first Python code file(txt\_extract.py) won’t run.
* “StopWords” folder: This folder contains a series of text files that contains the stop words. Without this folder, certain variables, as needed by the assignment, can’t be calculated.
* “MasterDictionary” folder: This folder contains two text files, one contains a list of positive words and the other contain a list of negative words. This are necessary to compute variables such as “Positive Variables”, “Negative Variables”. Without this folder, these variables can’t be calculated.