**Approach**

1. **Web Scraping:**
   1. Initially started with scraping a single blog to find the **selectors** for the head and body.
   2. Checked if **headers** are giving any issue or any extra steps have to be taken to verify there are no errors connecting to the website and being able to select the head and body for each blog.
   3. Imported the CSV input file to get the URL\_ID and the links for each blog.
   4. Made a loop to iterate through each row, retrieved the text, and appended it to a Pandas DataFrame.
   5. Then saved the blogs in a **.txt file using URL\_ID** as the name for the file.
   6. Introducing a **try and except block** to manage different selectors due to some errors caused by changes in selectors.
   7. **Blogs 36 and 49** were **missing**: Used ‘page.status\_code()’ to only fetch websites if the code was equal to 200.
   8. Finally, call the functions to first create the article directory if it does not exist, then called the scraping function to save the files in it.
2. **Text Analysis**
   1. Imported the libraries and the combined data (data with blogs as rows from web scraping)
   2. Removed html special characters and numbers as numbers do not provide any value to sentimental analysis
   3. Tokenized the words using nltk
   4. Thought of using stemmer to convert all the words in their root form but that would not match the given stopwords
   5. Imported stopwords from the .txt files and merged them all together, this could have been done in a single loop, but all the files had different textual issues
   6. Extracted the Sentimental metrics using functions using the dictionary of positive and negative words
   7. Used an earlier set of **datasets** named ‘seo\_analysis’ which had all the words before removing stopwords and performed all the calculations
   8. Used self-made function and regular expressions to calculate the remaining metrics
   9. Finally **merged** all the different **datasets** on **URL\_ID** and extracted the required columns in correct order
3. **Possible Errors**:
   1. Used dataset with removed stopwords from given files and did not remove any stopwords using nltk for sentimental analysis
   2. Did not remove any stopwords for the Readability Analysis
   3. Only removed stopwords from ‘nltk stopwords’ for the metrics other than the two mentioned above
   4. Performed the text analysis in Jupyter notebook but later pasted it in .py file as it was specified that only .py file is needed

**Instructions**

1. **How to run the file:**
   1. **PRE:** Install are the dependencies specified below in requirements
   2. **STEP 1**: First run the **web\_scrape.py** file which will create a article directory to save all the blogs. It will also create a **combined.csv**  to save all the blogs in a single file for NLP
   3. **STEP 2**: Then run the **txt\_analysis.py** to perform all the analysis and output a csv file **output.csv** for the output
   4. **Files Needed**: The **txt\_analysis.py** needs the input.csv, MasterDictionary, StopWords and articles.
2. **Requirements**:
   1. pandas==1.5.3
   2. beautifulsoup4==4.12.2
   3. requests==2.28.2
   4. nltk==3.8.1
   5. syllapy==0.7.2