# *Explanation*

Certainly! Let's break down the text extraction and text analysis process, along with instructions on how to run the Python script:

## Text Extraction:

**1. Approach to Text Extraction:**

- Utilized the `pandas` library to read data from an Excel file named "input.xlsx," containing URLs and corresponding URL_IDs.
- Applied web scraping techniques using `requests` and `BeautifulSoup` to extract article text from each URL.
- Saved the extracted text into individual text files, naming them based on their corresponding URL_IDs.

**2. Dependencies for Text Extraction:**

- `pandas`: For reading data from Excel.
- `requests`: For making HTTP requests to URLs.
- `BeautifulSoup`: For parsing HTML content and extracting text.

## Text Analysis:

**1. Approach to Text Analysis:**

- Cleaned the extracted text by removing stop words using the NLTK library.
- Calculated readability metrics, including average sentence length, percentage of complex words, and Fog Index.
- Utilized the `syllables` library to estimate syllable counts in words.
- Conducted sentiment analysis using `TextBlob`, calculating polarity and subjectivity scores.
- Determine personal pronoun counts using regular expressions.
- Computed additional metrics such as average word length, complex word count, and others.

**2. Dependencies for Text Analysis:**

- `pandas`: For data manipulation and storage.
- `nltk`: For natural language processing tasks such as tokenization, stop-word removal, and sentence splitting.
- `textblob`: For sentiment analysis.
- `syllables`: For estimating syllable counts in words.

## Instructions to Run the .py File:

**1. Install Dependencies:**

- Ensure that you have Python installed on your system.

- Open a terminal or command prompt.
- Run the following command to install the required dependencies:

```
pip install pandas nltk textblob syllables
```

**2. Prepare Input Data:**
- Place the "input.xlsx" file in the same directory as the Python script.
- The "input.xlsx" file should contain columns "URL_ID" and "URL" with the relevant data.

**3. Run the Script:**
- Save the provided Python script in a file, e.g., `text_analysis_script.py`.
- Open a terminal or command prompt.
- Navigate to the directory containing the script.
- Run the script using the following command:

```
python text_analysis_script.py
```

**4. View Output:**
- The script will generate an Excel file named "TextAnalysisOutput.xlsx" in the same directory.
- This file will contain the computed text analysis variables for each article.

By following these instructions, you should be able to run the Python script successfully and generate the desired output. Ensure that the required input file is present, and dependencies are installed before running the script.