```
Text summarization using beautiful soaf  and spacy from web link even it
doesent have permit to acces

```

In [23]:

```python
#import all libraries
import requests
from bs4 import BeautifulSoup
import spacy
from collections import Counter
```

In [24]:

```python
# define the fetch function fetching the web conent and get acces denying restri

def fetch_website_content(url):
    headers = {
        "User-Agent": "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537
    }
    response = requests.get(url, headers=headers)

    if response.status_code == 200:
        return response.text
    else:
        print(f"Failed to fetch content. Status code: {response.status_code}")
        return None

```

In [25]:

```python
#define parse  html function
def parse_html(html_content):
    soup = BeautifulSoup(html_content, 'html.parser')
    # Assuming the main content is inside <p> or <div> tags
    main_content = soup.find_all(['p', 'div'])
    return " ".join([p.get_text() for p in main_content])
```

In [26]:

```python
#Define the text summarization function
def text_summarization(text, num_sentences=3):
    nlp = spacy.load("en_core_web_sm")
    doc = nlp(text)
    # Remove stop words and punctuation
    word_frequencies = Counter(token.text.lower() for token in doc if not token.
    # Calculate sentence ranks based on word frequencies
    sentence_ranks = {sentence: sum(word_frequencies[token.text.lower()] for tok
    # Sort sentences by rank and select top sentences
    top_sentences = sorted(sentence_ranks, key=sentence_ranks.get, reverse=True)
    summary = " ".join(sentence.text for sentence in top_sentences)
    return summary
```

In [27]:

```python
# define the main function for first web link
def main():
    # Provide the URL of the website you want to scrape
    url = "https://searchengineland.com/guide/what-is-seo"
    html_content = fetch_website_content(url)
    if html_content:
        text_content = parse_html(html_content)
        num_sentences = 3  # Specify the number of sentences you want in the sum
        summary = text_summarization(text_content, num_sentences)
        print("Summary:")
        print(summary)
if __name__ == "__main__":
    main()
```

```
Marketing opportunities
Staff



Follow us

Facebook
Twitter
LinkedIn
Newsletters
RSS
YouTube



Topics

SEO
PPC
```

In [28]:

```python
# use the  the main function for second  web link
def main():
    # Provide the URL of the website you want to scrape
    url = "https://www.wordstream.com/seo"
    html_content = fetch_website_content(url)
    if html_content:
        text_content = parse_html(html_content)
        num_sentences = 3  # Specify the number of sentences you want in the su
        summary = text_summarization(text_content, num_sentences)
        print("Summary:")
        print(summary)
if __name__ == "__main__":
    main()
```

```
Solutions
Careers
Contact Us855.967.3787
+1.866.971.0933
    LearnBlog
Top Blog Posts
Guides
PPC University
Marketing 101
Infographics
Free ToolsGoogle Ads Grader
Facebook Ads Grader
Free Keyword Tool
Website Grader
 LearnBlog
Top Blog Posts
Guides
PPC University
Marketing 101
Infographics
```

In [ ]:

```python

```