

네, 일반적인 웹 크롤러 기능을 사용하여 엘라스틱 서치의 유료 리포트 기능을 사용하지 않고, 필요한 그래프 및 정보를 긁어와서 나만의 리포트를 만들 수 있습니다. 이 과정은 몇 가지 단계로 나눌 수 있으며, 필요한 도구와 기술을 활용하면 가능합니다.

## 1. 웹 크롤러 설정

먼저, 필요한 데이터가 있는 웹페이지를 크롤링해야 합니다. 이를 위해 웹 크롤러를 설정합니다. 웹 크롤러는 웹페이지의 내용을 자동으로 탐색하고 필요한 데이터를 수집하는 스크립트 또는 프로그램입니다. Python의 **BeautifulSoup**, **Scrapy**, **Selenium**과 같은 라이브러리를 사용할 수 있습니다.

예를 들어, **BeautifulSoup**와 **requests** 라이브러리를 사용하여 특정 웹페이지의 그래프나 표 데이터를 가져올 수 있습니다.

```
import requests
from bs4 import BeautifulSoup

url = 'http://example.com/data-page'
response = requests.get(url)
soup = BeautifulSoup(response.text, 'html.parser')

# 특정 HTML 요소(예: 테이블 또는 그래프 데이터)를 선택하여 가져오기
table = soup.find('table', {'id': 'data-table'})
rows = table.find_all('tr')

for row in rows:
    cols = row.find_all('td')
    data = [col.text for col in cols]
    print(data)
```

이 코드는 특정 웹페이지의 테이블 데이터를 가져오는 예시입니다. 페이지의 HTML 구조에 따라 코드를 조정해야 합니다.

## 2. 데이터 전처리

웹 크롤러로 수집한 데이터는 다양한 형식(예: HTML, JSON, CSV 등)으로 존재할 수 있습니다. 이 데이터를 전처리해야 합니다. 전처리는 데이터를 분석에 적합한 형태로 변환하는 과정입니다. 예를 들어, 날짜 형식을 통일하거나 불필요한 데이터를 제거하는 등의 작업이 포함됩니다.

Python의 **pandas** 라이브러리는 데이터 정리에 매우 유용합니다.

```
import pandas as pd

# 크롤링한 데이터가 포함된 리스트 또는 딕셔너리를 데이터프레임으로 변환
data = {'Column1': ['value1', 'value2'], 'Column2': ['value3', 'value4']}
df = pd.DataFrame(data)

# 필요한 전처리 작업 수행
df['Column1'] = df['Column1'].str.strip() # 문자열 공백 제거
```

### 3. 데이터 시각화

전처리된 데이터를 바탕으로 시각화를 할 수 있습니다. Python의 **matplotlib** 또는 **seaborn** 같은 시각화 라이브러리를 사용하여 그래프를 그립니다.

```
import matplotlib.pyplot as plt

# 데이터 시각화
plt.plot(df['Column1'], df['Column2'])
plt.title('My Data Visualization')
plt.xlabel('X Axis')
plt.ylabel('Y Axis')
plt.show()
```

### 4. 리포트 생성

수집하고 시각화한 데이터를 바탕으로 나만의 리포트를 만들 수 있습니다. 이 리포트는 단순한 PDF 파일이 될 수도 있고, 웹 대시보드 형태로 만들 수도 있습니다. `matplotlib`나 `seaborn`으로 생성한 그래프를 HTML 파일로 저장하거나, PDF로 내보낼 수 있습니다.

PDF 리포트를 생성하려면 `matplotlib`의 `savefig` 메서드를 사용할 수 있습니다.

python

코드 복사

```
plt.savefig('report.pdf') # 그래프를 PDF 파일로 저장
```

또는 웹 기반 리포트를 만들기 위해 `Dash`나 `Flask`와 같은 웹 프레임워크를 사용할 수 있습니다. 이렇게 하면 웹브라우저에서 데이터를 실시간으로 시각화하고 대시보드를 구축할 수 있습니다.

## 5. 자동화

위의 과정을 자동화하면, 주기적으로 새로운 데이터를 가져와서 업데이트된 리포트를 생성할 수 있습니다. 이를 위해 크론 작업(Cron Job)이나 다른 스케줄러를 사용해 파이썬 스크립트를 정기적으로 실행할 수 있습니다.

## 결론

일반적인 웹 크롤러와 `Python`의 다양한 도구를 사용하면 엘라스틱 서치의 유료 리포트 기능을 사용하지 않고도 원하는 데이터를 수집하고 시각화하여 나만의 맞춤형 리포트를 만들 수 있습니다. 이러한 접근 방식은 비용을 절감하면서도 유연하게 데이터를 관리하고 분석할 수 있는 방법입니다. 다만, 웹 크롤링은 대상 사이트의 이용 약관을 준수해야 하며, 법적 및 윤리적 규정을 지켜야 합니다.