

Python Aufgabe - Webscraping & Artikeltitel speichern

Dozent: Navjit Singh

Ziel: Webseiten analysieren, Inhalte automatisiert auslesen und lokal speichern.

Aufgabenstellung

Du willst regelmässig Nachrichtenartikel oder Blogeinträge analysieren. Dazu soll ein Python-Skript automatisch eine Webseite aufrufen, die Überschriften (z. B. Artikel-Titel) herausfiltern und lokal abspeichern.

Anforderungen

- Verwende `requests`, um eine Webseite (z. B. eine Nachrichten- oder Tech-Seite) aufzurufen.
- Verwende `BeautifulSoup`, um alle `<h2>`- oder `<a>`-Titel auf der Seite zu extrahieren.
- Speichere die gefundenen Titel zeilenweise in einer Textdatei `titel.txt`.
- Bonus: Prüfe, ob bestimmte Schlagwörter (wie 'KI', 'Python', 'Technologie') vorkommen.

Hinweise

- Nutze `pip install requests beautifulsoup4` zur Vorbereitung.
- Achte darauf, keine Seiten zu scrapen, die Scraping verbieten (z. B. viele News-Portale).
- Als Beispielseite eignet sich: <https://realpython.com/>

Beispielcode (Startgerüst)

```
import requests
from bs4 import BeautifulSoup

url = "https://realpython.com/"
response = requests.get(url)
soup = BeautifulSoup(response.text, "html.parser")

titles = soup.find_all("h2")
with open("titel.txt", "w", encoding="utf-8") as file:
    for title in titles:
        file.write(title.get_text(strip=True) + "\n")
```

Vertiefungsideen

- Extrahiere Links (``) zu den Artikeln

- Lade automatisch auch Unterseiten (Pagination)
- Speichere Ergebnisse in CSV statt TXT