untitled0

November 5, 2023

Nama : Kyla Belva Queena

NIM : 164221015

Kelas : SD-A1

Tugas : Praktikum TM 8

#**No. 1**#

Dengan menggunakan Python Requests dan BeautifulSoup cobalah untuk scraping 1 halaman website unair news (https://unair.ac.id/news) dan judul berita yang ada hari ini

[ ]:
```
import requests
response = requests.get('https://unair.ac.id/news')

response.text
```

Sebelumnya kita menginstall beberapa modul di cmd maupun di python, yaitu dengan py -m pip install [...]. Modul yang saya download adalah request, beautifulsoup4, pandas, scrapy, dan nbconvert.

1. Melakukan impor modul "requests" ke dalam program Python. Request digunakan untuk melakukan permintaan HTTP seperti mengambil halaman web.
2. Menggunakan modul "requests" untuk mengirim permintaan HTTP GET ke URL 'https://unair.ac.id/news'. Hasilnya disimpan dalam variabel response. Variabel ini nantinya berisi objek respons HTTP.
3. Mengakses teks/konten dari respons HTTP.

–> mengambil isi HTML dari web unair

[ ]:
```
from bs4 import BeautifulSoup

response = requests.get('https://unair.ac.id/news')
rawhtml = response.text
soup = BeautifulSoup(rawhtml, 'html.parser')
soup
```

1. Melakukan import modul 'BeautifulSoup'. Modul ini dipakai untuk melakukan analisis dan ekstraksi data dari dokumen HTML.

2. Mengirim permintaan HTTP GET ke URL ‘https://unair.ac.id/news’ dan menyimpan responsnya dalam variabel response.
3. Mengambil teks HTML dari respons HTTP yang diterima dan menyimpannya dalam variabel rawhtml.
4. Membuat objek BeautifulSoup.

’rawhtml : teks HTML yang akan dianalisis

‘html.parser’ : parser HTML yang digunakan untuk mengurai dokumen HTML

5. Memanggil dan menampilkan objek ‘soup’ yang berisi struktur data yang dapat digunakan untuk mengekstraksi informasi dari halaman web.

–> mengekstrak dan mencari data dari HTML

```
[ ]: for i in soup.find_all('h2'):
       if i.find_next('span',{'class' : 'elementor-post-date'}).text.strip() == '03/
     ↪11/2023' :
         print(i.get_text())
```

1. Melakukan for looping. Loop ini mengiterasi melalui semua elemen HTML dengan tag h2 yang ditemukan dalam objek soup. Semua elemen h2 disimpan dalam variabel i untuk setiap iterasi.
2. Mencari elemen span yang memiliki atribut class dengan nilai elementor-post-date yang berdekatan dengan elemen h2 dan mengambil teks dari elemen tersebut. Selain itu text.strip untuk memastikan bahwa teks tersebut adalah ‘05/11/2023’. Jika terpenuhi, maka blok kode dibawahnya akan dieksekusi.
3. Jika tanggal sesuai, program akan mencetak teks yang terdapat dalam elemen h2 yang ditemukan dalam iterasi saat ini.

#**No. 2**#

Lalu, lakukan crawling unair news dengan mengkombinasikan Python Request, beautifulsoup, dan for loop, untuk mengambil judul berita dari kategori featured news

```
[ ]: import requests
     from bs4 import BeautifulSoup

     for i in range(1, 4):
         url = "https://unair.ac.id/category/featured/page/"+str(i)+"/"
         response = requests.get(url)
         rawhtml = response.text
         soup = BeautifulSoup(rawhtml,'html.parser')
         for j in soup.find_all('h3'):
             print(j.get_text())
```

1. Melakukan import request untuk melakukan permintaan HTTP ke situs web.
2. Melakukan import beautifulsoup dari bs untuk melakukan analisis HTML pada halaman web yang diunduh.

3. Loop for yang akan mengiterasi tiga kali, dengan nilai i dari 1 hingga 3. Kode akan mengunjungi tiga halaman web yang berbeda.
4. Pada setiap iterasi, URL halaman web yang akan diunduh dibentuk dengan menggabungkan URL dasar dengan nomor halaman yang sesuai. Nomor halaman diubah menjadi string dan ditambahkan ke URL.
5. Membuat variabel response untuk menyimpan hasil respon permintaan http get ke url yang dibentuk
6. Mengambil teks html dari respon http dan disimpan dalam variabel rawhtml1
7. Membuat objek BeautifulSoup untuk menganalisis teks HTML yang telah diunduh. Objek ini digunakan untuk mencari elemen di dalam HTML.
8. Mencari semua elemen h3 dalam halaman web yang diuraikan oleh variabel soup
9. Melakukan print untuk mencetak judul-judul yang ada di halaman web tersebut.

#**No. 3**#

Setelah itu, gunakan scrapy untuk melakukan crawling website https://bit.ly/scrapingtry dan menyimpan judul game beserta harganya dalam tabel

[ ]:
```
pip install scrapy
```

Melakukan installasi modul scrapy dalam python

[ ]:
```
import scrapy
import pandas as pd

class PlaystationSpider(scrapy.Spider):
  name = 'scrape_ps'
  base_url = 'https://store.playstation.com/en-id/category/
↪05a2d027-cedc-4ac0-abeb-8fc26fec7180/'

  def start_requests(self):
    num_pages = 7
    for page in range(1, num_pages+1):
      url = f'{self.base_url}{page}/'
      yield scrapy.Request(url, callback+self.parse)

  def parse(self, response):
    titles = response.css('span.psw-t-body.psw-c-t-1.psw-t-truncate-2.psw-m-b-2:
↪:text').getall()
    prices = response.css('span.psw-m-r-3::text').getall()

    for title, price in zip(titles, prices):
      yield {
          'title' : title.strip(),
          'price' : price.strip()
      }
```

1. Import modul scrapy. Modul ini dipakai untuk melakukan web scraping
2. Import modul pandas. Modul ini untuk mengelola dan menyimpan data
3. Membuat kelas PlaystationSpider. Kelas ini merupakan subclass dari scrapy.spider. Di kelas ini adalah tempat melakukakn scraping
4. Membuat variabel name yang memberi nama untuk spider ini
5. Membuat vaiabel base_url untuk menyimpan url dasar yang digunakan sebagai basis untuk mengambil data
6. Membuat fungsi start_requests(self). Disinilah dimana program memulai permintaan pertama untuk mengambil halaman web. Nantinya akan menghasilkan URL untuk beberapa halaman yang akan di scrape
7. Membuat variabel num_pages untuk menentukan jumlah halaman yang discrape. Disini saya memilih 7 halaman
8. Membuat for looping untuk menghasilkan url tiap halaman yang akan discrape (halaman 1-7)
9. Menggabungkan nomor halaman dengan base_url untuk membentuk URL halaman yang akan discrape
10. Mengirimkan permintaan ke URL halaman yang akan discrape juga menetapkan fungsi parse sebagai callback yang akan memproses respons
11. Membuat fungsi parse yang akan dipanggil untuk memproses respons dari halaman web
12. Menggunakan css selector untuk mengambil teks dari HTML yang sesuai dengan kriteria yang diberikan lalu menyimpannya dalam variabel titles
13. Menggunakan CSS Selector untuk mengambil teks dari elemen HTML yang sesuai dengan kriteria yang diberikan lalu menyimpannya dalam variabel prices
14. Menggunakan for loop untuk iterasi melalui titles dan prices secara bersamaan
15. Untuk menghasilkan dictionary untuk setiap pasangan title dan price, dengan menghapus spasi ekstra menggunakan metode strip().

–> melakukan web scraping pada halaman web PlayStation Store, mengambil judul dan harga game, dan menyimpannya dalam format yang sesuai.

#**No. 4**#

Simpan file hasil crawling dalam Repo github masing masing. (buat repo baru khusus scraping, Bisa dalam Json / csv)

Membuat repository baru pada github, disini saya beri nama tm8_scrapy. Lalu mengupload csv, csv, dan lainnya

LINK GITHUB : https://github.com/kylaqueena7/tm8_scrapy