

LAPORAN TUGAS INDIVIDU 3

DATA SCRAPING

Disusun untuk Memenuhi Tugas Individu 3 Mata Kuliah Data Scraping
Dibimbing oleh Pelsri Ramadar N.S. M.Kom



Oleh:

Sherly Puput Rahmawati (1123102147)

PROGRAM STUDI S1 TEKNIK INFORMATIKA
SEKOLAH TINGGI ILMU KOMPUTER PGRI BANYUWANGI
2024

Kode Program

```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
from bs4 import BeautifulSoup

URL = 'https://www.tokopedia.com/search?'
params = {
    'q': 'bagiak%20khas%20banyuwangi'
}

headers = {'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/131.0.0.0 Safari/537.36'}

driver = webdriver.Chrome()

fullURL = f"{URL}&q={params['q']}"
driver.get(fullURL)

WebDriverWait(driver, 10).until(EC.presence_of_element_located((By.CLASS_NAME, 'css-5wh65g'))))

html = driver.page_source

soup = BeautifulSoup(html, 'html.parser')

data = soup.find_all('div', {'class': 'css-5wh65g'})

for i in range(len(data)):
    nama = data[i].find('div', {'class': '_6+0pBPVGAqgnmycna+bWIw=='})
    harga = data[i].find('div', {'class': '_67d6E1xDKIzw+i2D2L0tjw=='})
    penjual = data[i].find('span', {'class': 'T0rpy-LEwYNQifsgB-3SQw=='})

    if nama and harga and penjual:
        print("Nama produk: " + nama.text.strip())
        print("Harga: " + harga.text.strip())
        print("Penjual: " + penjual.text.strip(), "\n")

driver.quit()
```

Penjelasan Kode Program:

Kode program ini digunakan untuk mengambil data produk "bagiak khas Banyuwangi" dari situs Tokopedia secara otomatis menggunakan Selenium dan BeautifulSoup. Program dimulai dengan mengimpor pustaka yang dibutuhkan untuk mengontrol browser dan membaca data

web. URL pencarian disiapkan dengan menambahkan parameter kata kunci `bagiak%20khas%20banyuwangi`. Selenium membuka browser Chrome dengan `webdriver.Chrome()` dan mengakses URL tersebut menggunakan `driver.get(fullURL)`. Program kemudian menunggu hingga elemen produk dengan class `css-5wh65g` muncul menggunakan `WebDriverWait`, dan memastikan halaman telah dimuat dengan benar. Setelah itu, isi halaman web diambil menggunakan `driver.page_source` dan dianalisis dengan BeautifulSoup. Program mencari semua elemen `<div>` dengan class `css-5wh65g`, yang berisi data produk. Untuk setiap produk yang ditemukan, program mencoba mengambil nama, harga, dan penjual dengan mencari elemen-elemen spesifik menggunakan `find`. Jika semua data tersedia, informasi produk dicetak ke terminal. Browser akhirnya ditutup dengan `driver.quit()`. Saat kode dijalankan, hasil yang muncul adalah daftar produk dengan nama, harga, dan penjual seperti "BAGIAK KUE KHAS BANYUWANGI (300gr)" seharga Rp25.000 dari penjual "kue kita 99" dan produk-produk lainnya dengan informasi yang serupa.

Hasil:

```
DevTools listening on ws://127.0.0.1:65427/devtools/browser/ad945b48-e8ca-4bac-99a9-7cfa388a9c6c
Nama produk: BAGIAK KUE KHAS BANYUWANGI (300gr)
Harga: Rp25.000
Penjual: kue kita 99

Nama produk: kue bagiak khas Banyuwangi kemasan dos
Harga: Rp19.000
Penjual: budzz store

Nama produk: Bagiak Irma Jaya makanan khas Banyuwangi
Harga: Rp19.000
Penjual: Rumah Cemilan.

Nama produk: Bagiak sherly khas banyuwangi
Harga: Rp30.000
Penjual: Rumah Hoqie Hoqie

Nama produk: kue bagiak khas Banyuwangi kemasan dos
Harga: Rp19.000
Penjual: larispuol

Nama produk: Bagiak makanan khas Banyuwangi (oreng osing)
Harga: Rp19.000
Penjual: Rumah Cemilan.

Nama produk: Kue Bagiak Khas Kota Banyuwangi - Rasa Keningar, Jahe dan Susu
Harga: Rp17.000
Penjual: Sangu Sore

Nama produk: ORENG OSING | Bagiak 500gr Khas Banyuwangi Terlaris.
Harga: Rp75.000
Penjual: Peache TK
```

Analisa Hasil

Program web scraping ini berhasil mengambil data produk "bagiak khas Banyuwangi" dari situs Tokopedia dengan menampilkan nama produk, harga, dan penjual. Meskipun proses berjalan sesuai harapan, terdapat duplikasi data karena produk yang sama muncul beberapa kali, kemungkinan akibat struktur halaman web. Selain itu, program sangat bergantung pada elemen HTML situs, sehingga perubahan kecil pada class elemen dapat menyebabkan kegagalan. Untuk meningkatkan fungsionalitas, disarankan menambahkan penghapusan duplikasi data, penanganan error, dan penyimpanan hasil ke dalam database agar lebih terorganisir dan mudah diakses.