**LAPORAN**

**RENCANA TUGAS MANDIRI (RTM) Ke-1**

**MATA KULIAH BIG DATA C**

**MERINGKAS MATERI KULIAH**



**DISUSUN OLEH:**

Mohamad Ibnu Fajar Maulana (21083010106)

**DOSEN PENGAMPU:**

Tresna Maulana Fahrudin S.ST., M.T. (NIP. 199305012022031007)

**PROGRAM STUDI SAINS DATA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN” JAWA TIMUR**

**2022**

# **DAFTAR ISI**

[**DAFTAR ISI** 2](#_Toc128509771)

[**1.1** **Definisi** 1](#_Toc128509772)

[**1.2** **Implementasi** 1](#_Toc128509773)

[**1.2.1** **Mendapatkan URL Google** 2](#_Toc128509774)

[**1.2.2** **Mendapatkan Teks Berita dari Google Lanjutan** 4](#_Toc128509775)

[**Lampiran** 7](#_Toc128509776)

# **Definisi**

*Web scraping* merupakan sebuah proses ekstrasi dimana data-data tersebut diambil dari situs internet/website. *Web scraping* bisa dilakukan dengan beberapa cara umum, diantaranya adalah :

* Manual : Cara manual dapat dilakukan apabila anda menyalin data dengan cara *copy paste* dari sebuah website.
* Otomatis : Cara otomatis dapat dilakukan dengan anda menggunakan koding atau ekstension browser.

Bukan hanya itu saja, *web scraping* dapat dilakukan dengan beberapa teknik, teknik tersebut ialah:

* Menggunakan reguler ekspression
* Parsing HTML
* Menganalisa DOM
* Menggunakan Xpath
* Menggunakan Google Sheet

# **Implementasi**

Pada laporan ini terdapat implementasi *web scraping* dimana pada studi kasus yang diangkat adalah membahas “**Sabda PS**” yang merupakan Chief of Education Officer sekaligus teacher of Zenius. Implementasi tersebut dapat anda lihat pada halaman selanjutnya.

## **Mendapatkan URL Google**

Pada codingan dibawah ini merupakan salah satu *web scraping* menggunakan url google. Dengan tipe file .py, kemudian lakukan dan jalankan kodingan seperto dibawah ini.

from bs4 import BeautifulSoup

from selenium import webdriver

from webdriver\_manager.chrome import ChromeDriverManager

chrome\_options = webdriver.ChromeOptions()

chrome\_options.add\_argument("--headless")

driver = webdriver.Chrome(ChromeDriverManager().install(), chrome\_options=chrome\_options)

#query ini untuk mendapatkan links

query = 'Sabda PS'

#links = [] #untuk menginisiasi list kosong untuk mendapatkan hasil akhir

#spesifik nomor dari halaman pencarian goggle, setiap page mengandung 10 link

n\_pages = 10

for page in range(1, n\_pages):

url = "https://www.google.com/search?q=" + query + "&start=" + str((page - 1)\*10)

driver.get(url)

soup = BeautifulSoup(driver.page\_source, 'html.parser')

#soup = BeautifulSoup(r.text, 'html.parser')

search = soup.find\_all('div', class\_="yuRUbf")

for h in search:

#link.append(h.a.get(href))

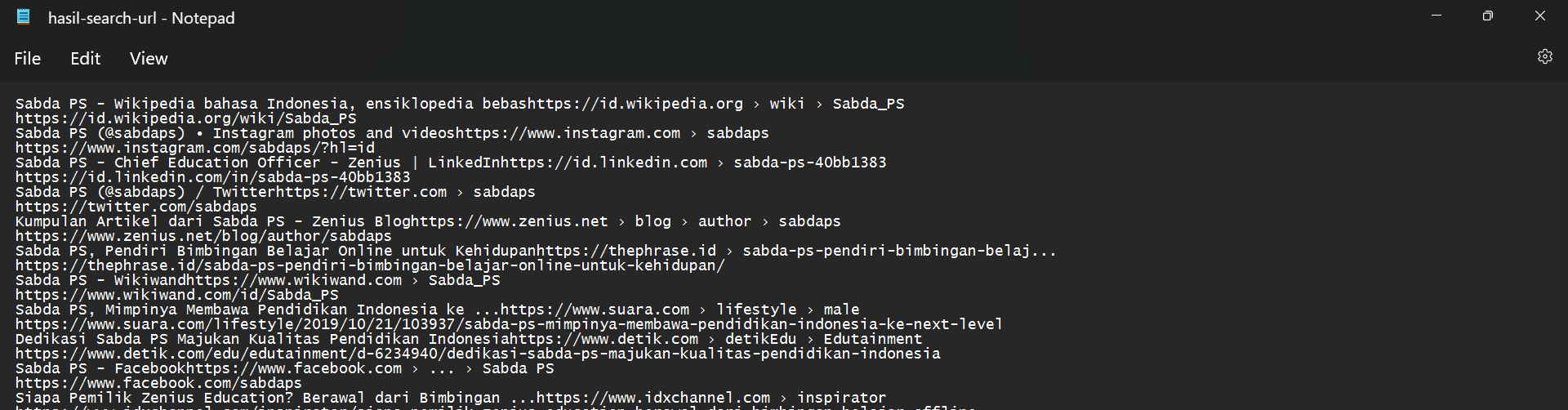
#print(h.a.text)

print(h.a.get('href'))

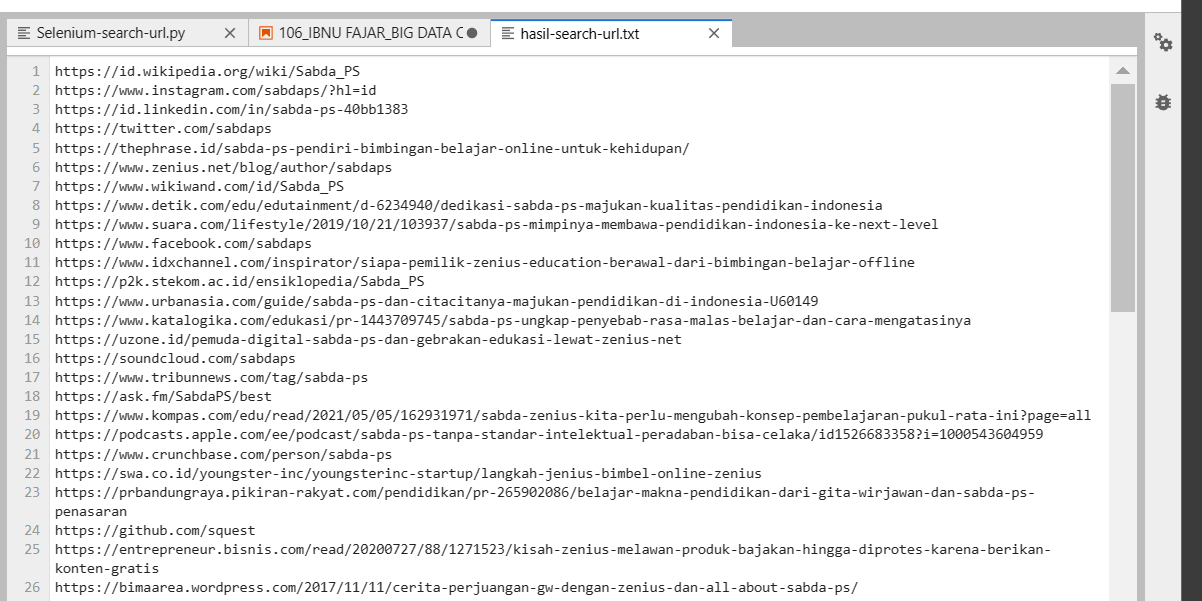
Kemudian setelah membuat kode skrip python pada file.py, lalu jalankan kodingan tersebut pada file.ipynb untuk menyimpan output pada file hasil-search-url.txt, kodingan tersebut terdapat dibawah ini:

!python Selenium-search-url.py > hasil-search-url.txt

Setelah itu, akan mendapatkan output hasil-search-url, seperti gambar dibawah ini:

****

Pada gambar dibawah ini dipersingkat jadinya hanya berisikan link atau urlnya saja.

****

## **Mendapatkan Teks Berita dari Google Lanjutan**

Pada codingan dibawah ini merupakan salah satu *web scraping* untuk mendapatkan teks lanjutan berita dari google. Dengan tipe file.py , kemudian lakukan dan jalankan kodingan seperti dibawah ini.

from selenium import webdriver

import re

import sys,getopt

import argparse

from webdriver\_manager.chrome import ChromeDriverManager

from selenium.webdriver.common.by import By

chrome\_options=webdriver.ChromeOptions()

chrome\_options.add\_argument('--headless')

driver=webdriver.Chrome(ChromeDriverManager().install(),chrome\_options=chrome\_options)

pattern = re.compile(r'\\\n')

full\_txt=[]

def parse\_args():

parser=argparse.ArgumentParser()

parser.add\_argument('-i','--infile',default='',help='input filename')

parser.add\_argument('-o','--outfile',default='',help='outfile filename')

return parser.parse\_args()

def main():

args=parse\_args()

outfile=args.outfile

infile=args.infile

with open(infile) as f:

content=f.read().splitlines()

f.close()

f=open(outfile,'w')

for u in content:

driver.get(u)

elemns=driver.find\_element(By.TAG\_NAME,'body').text

menyaring\_text=[line for line in elemns.split('\n')if not pattern.search(line)]

full\_txt.append(menyaring\_text)

import json

f.write(json.dumps(full\_txt))

driver.close()

f.close()

main()

Setelah menjalankan file.py, anda harus menjalankan file.ipynb untuk menyimpan hasil output pada file bernama hasil-browser.txt.

!python Selenium-browse-url.py -i hasil-search-url.txt -o hasil-browse-url.txt

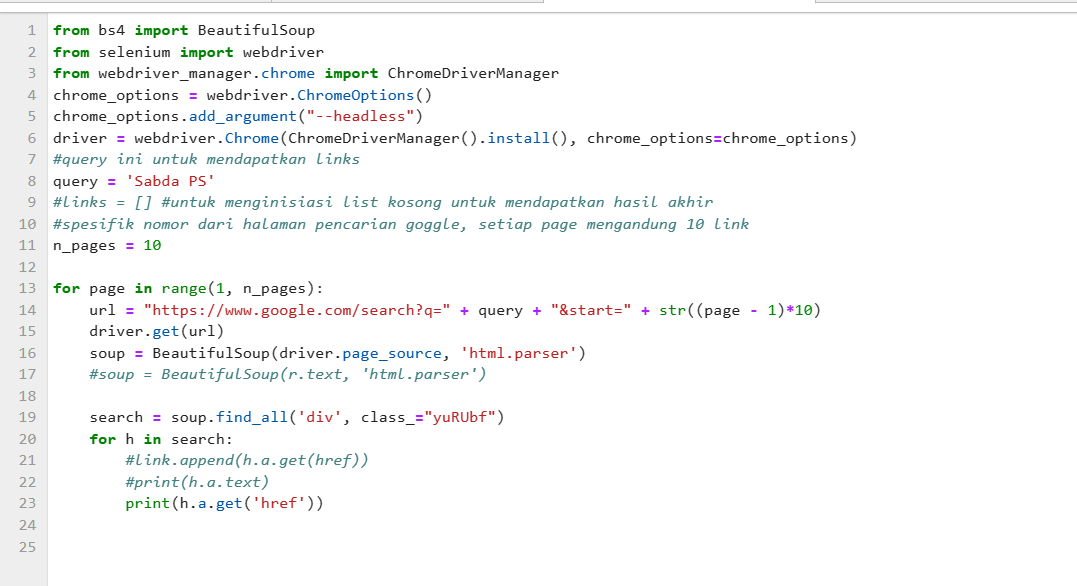
Lalu , setelah melakukan running code maka akan menghasilkan output sebagai berikut ini:

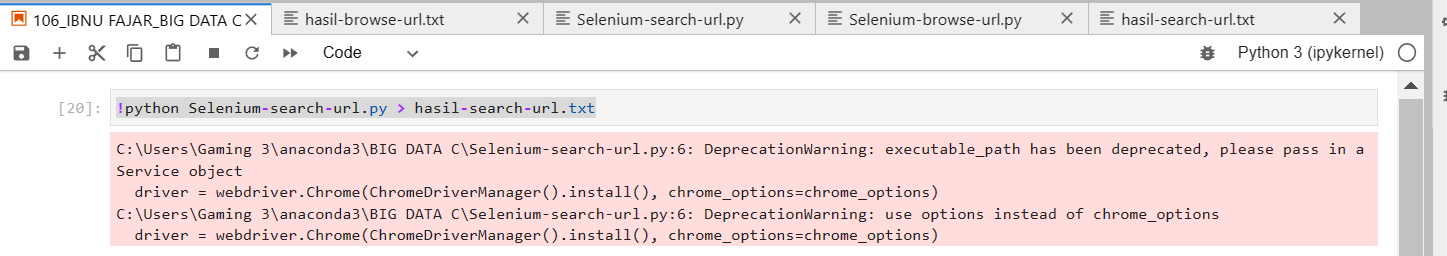


**Hasil output diatas sudah dilakukan filter text untuk menyaring beberapa kata yang tidak penting.**

# **Lampiran**

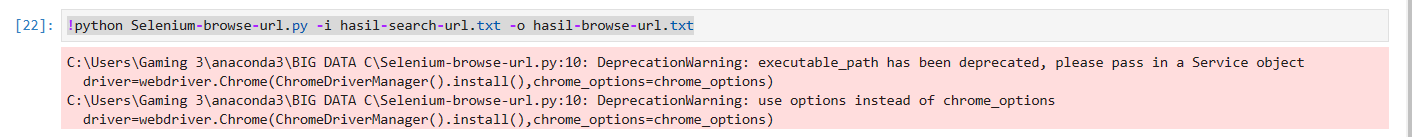
* Skrip kodingan mendapatkan url google



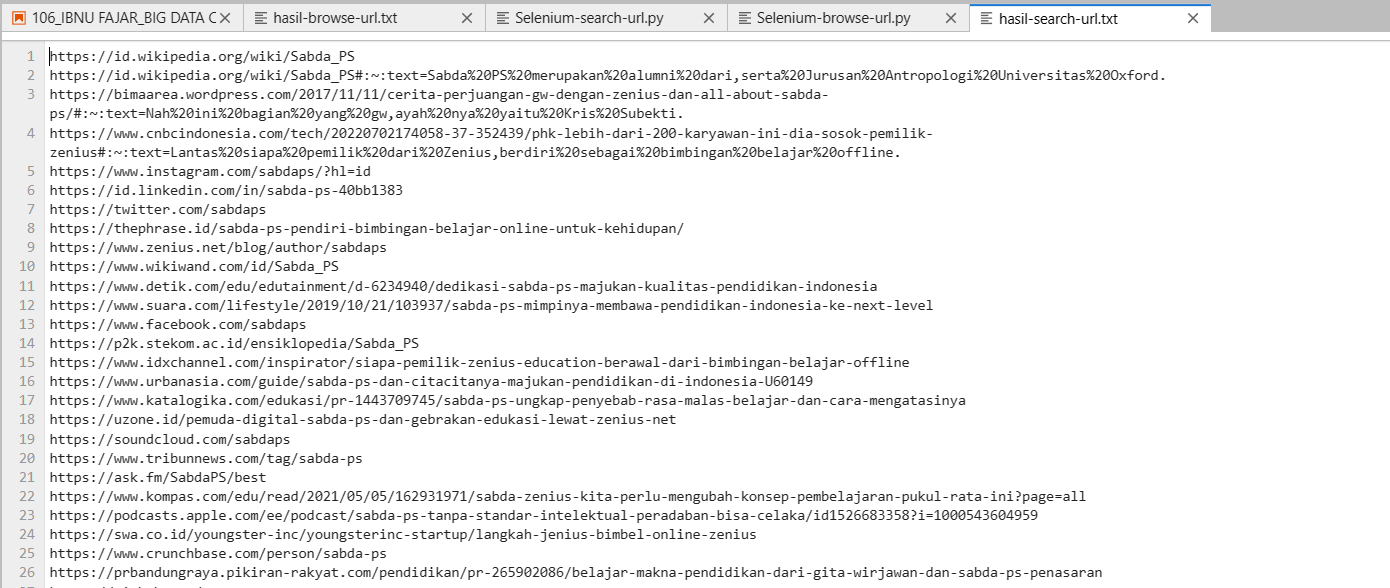


* Skrip kodingan mendapatkan teks berita dari google





* Output hasil-search



* Output hasil-browse

