Nama : Raihan Rahmanda Junianto

NIM : 222112303

Kelas : 3SD2

**Penugasan Praktikum 12 Information Retrieval**

Praktikum kali ini memiliki tujuan untuk melakukan scraping ulasan pengguna yang diperoleh melalui Google Maps pada tempat yaitu Jakarta Aquarium dengan link: <https://maps.app.goo.gl/SyXMj8rCRsco4JM38>.

Berikut disajikan keseluruhan kode untuk melalukan tugas tersebut.

"""This script serves as an example on how to use Python

   & Playwright to scrape/extract data from Google Maps"""

from playwright.sync\_api import sync\_playwright

from dataclasses import dataclass, asdict, field

import pandas as pd

import argparse

import time

import sys

@dataclass

class Review:

    """holds Review data"""

    id\_review: str = None

    name: str = None

    review\_text: str = None

@dataclass

class ReviewList:

    """holds list of Review objects,

    and save to both excel and csv

    """

    review\_list: list[Review] = field(default\_factory=list)

    def dataframe(self):

        """transform Review\_list to pandas dataframe

        Returns: pandas dataframe

        """

        return pd.json\_normalize((asdict(review) for review in self.review\_list), sep="\_")

    def save\_to\_excel(self, filename):

        """saves pandas dataframe to excel (xlsx) file

        Args:

            filename (str): filename

        """

        self.dataframe().to\_excel(f"{filename}.xlsx", index=False)

    def save\_to\_csv(self, filename):

        """saves pandas dataframe to csv file

        Args:

            filename (str): filename

        """

        self.dataframe().to\_csv(f"{filename}.csv", index=False)

def main():

    with sync\_playwright() as p:

        browser = p.chromium.launch(headless=False)

        page = browser.new\_page()

        page.goto("https://maps.app.goo.gl/SyXMj8rCRsco4JM38", timeout=60000)

        page.wait\_for\_timeout(5000)

        page.locator('button:has-text("Ulasan lainnya")').click();

        review\_list = ReviewList()

        page.wait\_for\_timeout(5000)

        for i in range(1, total):

            review = Review()

            review\_element = page.query\_selector('//\*[@id="QA0Szd"]/div/div/div[1]/div[2]/div/div[1]/div/div/div[2]/div[9]/div['+str(3\*i+-2)+']/div/div/div[2]/div[2]/div[1]/button')

            review\_id = review\_element.get\_attribute('data-review-id')

            reviewer\_name\_xpath = '//\*[@id="QA0Szd"]/div/div/div[1]/div[2]/div/div[1]/div/div/div[2]/div[9]/div['+str(3\*i+-2)+']/div/div/div[2]/div[2]/div[1]/button/div[1]'

            reviewer\_name = page.locator(reviewer\_name\_xpath).inner\_text()

            review\_xpath = '//\*[@id="'+review\_id+'"]'

            if "… Lainnya" in page.locator(review\_xpath).inner\_text():

                button\_lainnya\_xpath = '//\*[@id="'+review\_id+'"]/span[2]/button'

                page.locator(button\_lainnya\_xpath).click();

            review\_text = page.locator(review\_xpath).inner\_text()

            review.id\_review = review\_id

            review.name = reviewer\_name

            review.review\_text = review\_text

            review\_list.review\_list.append(review)

            page.mouse.wheel(0, 7000)

            page.wait\_for\_timeout(3000)

            # Print 14 huruf terakhir review id dan total review yang sudah di scrape

            print(f"Review ID: ...{review\_id[-14:]} | Currently Scraped: {i}", end='\r')

            sys.stdout.flush()

        review\_list.save\_to\_excel("D:/RAIHAN STIS/Perkuliahan/SEMESTER 5/Praktikum INFORMATION RETRIEVAL/Pertemuan (12)/jakarta\_aquarium\_review\_data")

        browser.close()

if \_\_name\_\_ == "\_\_main\_\_":

    parser = argparse.ArgumentParser()

    parser.add\_argument("-t", "--total", type=int)

    args = parser.parse\_args()

    # total number of products to scrape. Default is 10

    if args.total:

        total = args.total + 1

    else:

        total = 20

    main()

Pada kode di atas menggunakkan beberapa library serta modul baru dengan masing-masing kegunaan sebagai berikut.

* ‘sync\_playwright’ dari modul ‘playwright.sync\_api’ berisi modul Playwright yang menyediakan antarmuka sinkron untuk otomatisasi browser.
* ‘dataclass, asdict, field’ dari modul ‘dataclasses’ digunakan untuk membuat kelas data (data class). Dalam konteks ini ditunjukkan oleh kelas ReviewList.
* ‘argparse’ digunakan untuk mengolah argumen baris perintah. Dalam konteks ini digunakan untuk mengatur argument pada fungsi main.
* ‘time’ digunakan untuk menangani waktu atau penundaan, seperti timeout, dan lain sebagainya.
* ‘sys’ merupakan modul sistem untuk interaksi dengan interpreter Python. Dalam hal ini, digunakan untuk menangani buffer output menggunakkan perintah stdout.

A screen shot of a computer code

Description automatically generated

Selain itu, terdapat juga fungsi ‘dataframe(self)’ yang digunakan untuk mengubah Review\_list menjadi sebuah dataframe berbasis pandas serta fungsi ‘save\_to\_excel’ dan ‘save\_to\_csv’ untuk menyimpan output scraping ke dalam dua format file, yaitu .xlsx dan .csv.

A screen shot of a computer program

Description automatically generated

Pada fungsi main, terdapat beberapa perintah yang digunakan untuk melakukan iterasi pengambilan ulasan pengguna, menampilkan progress scraping, hingga menyimpan output scraping ke dalam file berekstensi .xlsx.

def main():

    with sync\_playwright() as p:

        browser = p.chromium.launch(headless=False)

        page = browser.new\_page()

        page.goto("https://maps.app.goo.gl/SyXMj8rCRsco4JM38", timeout=60000)

        page.wait\_for\_timeout(5000)

        page.locator('button:has-text("Ulasan lainnya")').click();

        review\_list = ReviewList()

        page.wait\_for\_timeout(5000)

        for i in range(1, total):

            review = Review()

            review\_element = page.query\_selector('//\*[@id="QA0Szd"]/div/div/div[1]/div[2]/div/div[1]/div/div/div[2]/div[9]/div['+str(3\*i+-2)+']/div/div/div[2]/div[2]/div[1]/button')

            review\_id = review\_element.get\_attribute('data-review-id')

            reviewer\_name\_xpath = '//\*[@id="QA0Szd"]/div/div/div[1]/div[2]/div/div[1]/div/div/div[2]/div[9]/div['+str(3\*i+-2)+']/div/div/div[2]/div[2]/div[1]/button/div[1]'

            reviewer\_name = page.locator(reviewer\_name\_xpath).inner\_text()

            review\_xpath = '//\*[@id="'+review\_id+'"]'

            if "… Lainnya" in page.locator(review\_xpath).inner\_text():

                button\_lainnya\_xpath = '//\*[@id="'+review\_id+'"]/span[2]/button'

                page.locator(button\_lainnya\_xpath).click();

            review\_text = page.locator(review\_xpath).inner\_text()

            review.id\_review = review\_id

            review.name = reviewer\_name

            review.review\_text = review\_text

            review\_list.review\_list.append(review)

            page.mouse.wheel(0, 7000)

            page.wait\_for\_timeout(3000)

            # Print 14 huruf terakhir review id dan total review yang sudah di scrape

            print(f"Review ID: ...{review\_id[-14:]} | Currently Scraped: {i}", end='\r')

            sys.stdout.flush()

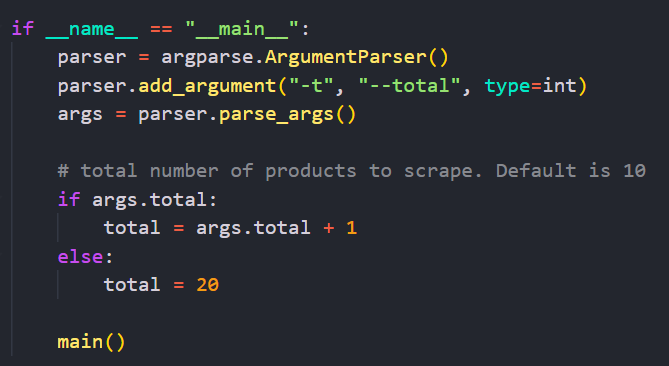
        review\_list.save\_to\_excel("D:/RAIHAN STIS/Perkuliahan/SEMESTER 5/Praktikum INFORMATION RETRIEVAL/Pertemuan (12)/jakarta\_aquarium\_review\_data")

        browser.close()

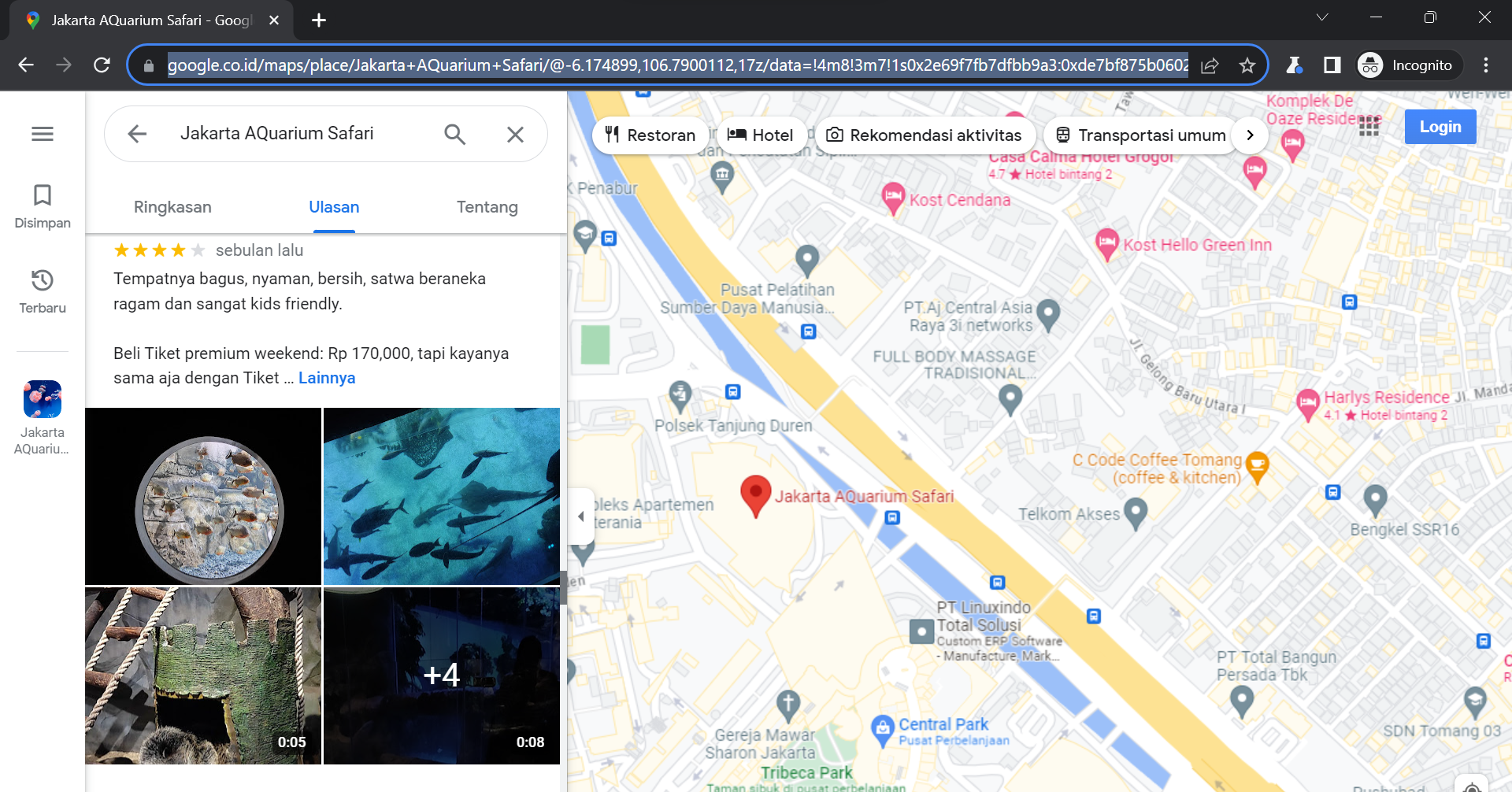
Pada iterasi pengambilan ulasan pengguna, terdapat perintah ‘page.mouse.wheel(0, 7000)’ yang digunakan untuk mengatur gulir mouse secara otomatis. Nilai 0 menyatakan jumlah gulir secara horizontal dan nilai 7000 menyatakan jumlah gulir vertical sebanyak 7000 unit. Jumlah gulir vertical dapat disesuaikan dengan jumlah ulasan yang akan discraping. A screen shot of a computer code

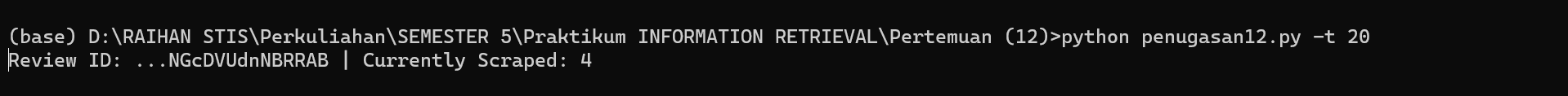
Description automatically generated

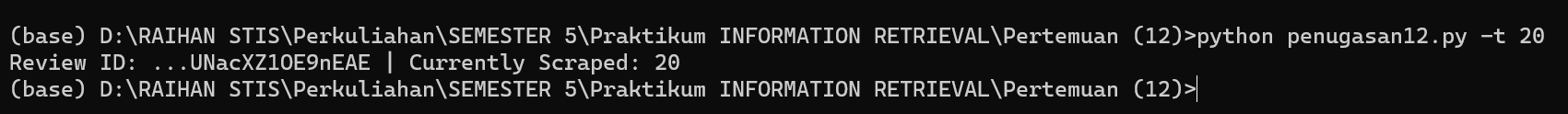
Program kali ini diatur untuk melakukan scraping total ulasan pengguna sebanyak 20 buah sehingga perlu dilakukan pendefinisian argument pada fungsi main.

­ 

Berikut merupakan tampilan output ketika kode program dijalankan.







A close-up of a text

Description automatically generated