Contents

1. Sử dụng Open Street Map - request

requestt.py > ...

```python
import requests
import time
import pandas as pd

def get_places_of_interest(latitude, longitude):
    url = f"https://nominatim.openstreetmap.org/reverse"
    params = {
        "format": "json",
        "lat": latitude,
        "lon": longitude,
        "addressdetails": 1,
    }

    response = requests.get(url, params=params)
    data = response.json()
    places_of_interest = []
    for key in ['building', 'hospital', 'factory','apartments','office','residential','aeroway
        if key in data['address']:
            places_of_interest.append(data['display_name'])
            return places_of_interest
        else:
            continue



df = pd.read_csv("4g.csv")
address1 = []
latitude = df["LATITUDE"]
longitude = df["LONGITUDE"]
# Nhập vào tọa độ latitude và longitude
for i in range (0,len(latitude)):
    print(i)
    places_of_interest = get_places_of_interest(latitude[i], longitude[i])
    if places_of_interest:
        address1.append(places_of_interest[0])

    else:
        address1.append(places_of_interest)
print("_____")
print("Completed")
df["Address1"] = address1
# Tên tệp Excel là 'output.xlsx', index=False để không bao gồm cột index
df.to_excel('4g_output1_use_libRequest.xlsx', index=False)

```

ở dòng 17 để chúng ta có thể chọn được các vị trí mong muốn

chúng ta có thể tham khảo ở link này : https://taginfo.openstreetmap.org/

đầu vào là file 4g.csv

đầu ra là file 4g_output1_use_libRequest . các tọa độ mà ko lấy được thông tin về sẽ để Null

Sau khi lọc ta sẽ được 2 file :

4g_output1_use_request_win

4g_output1_use_request_fail

filter_request.py > ...

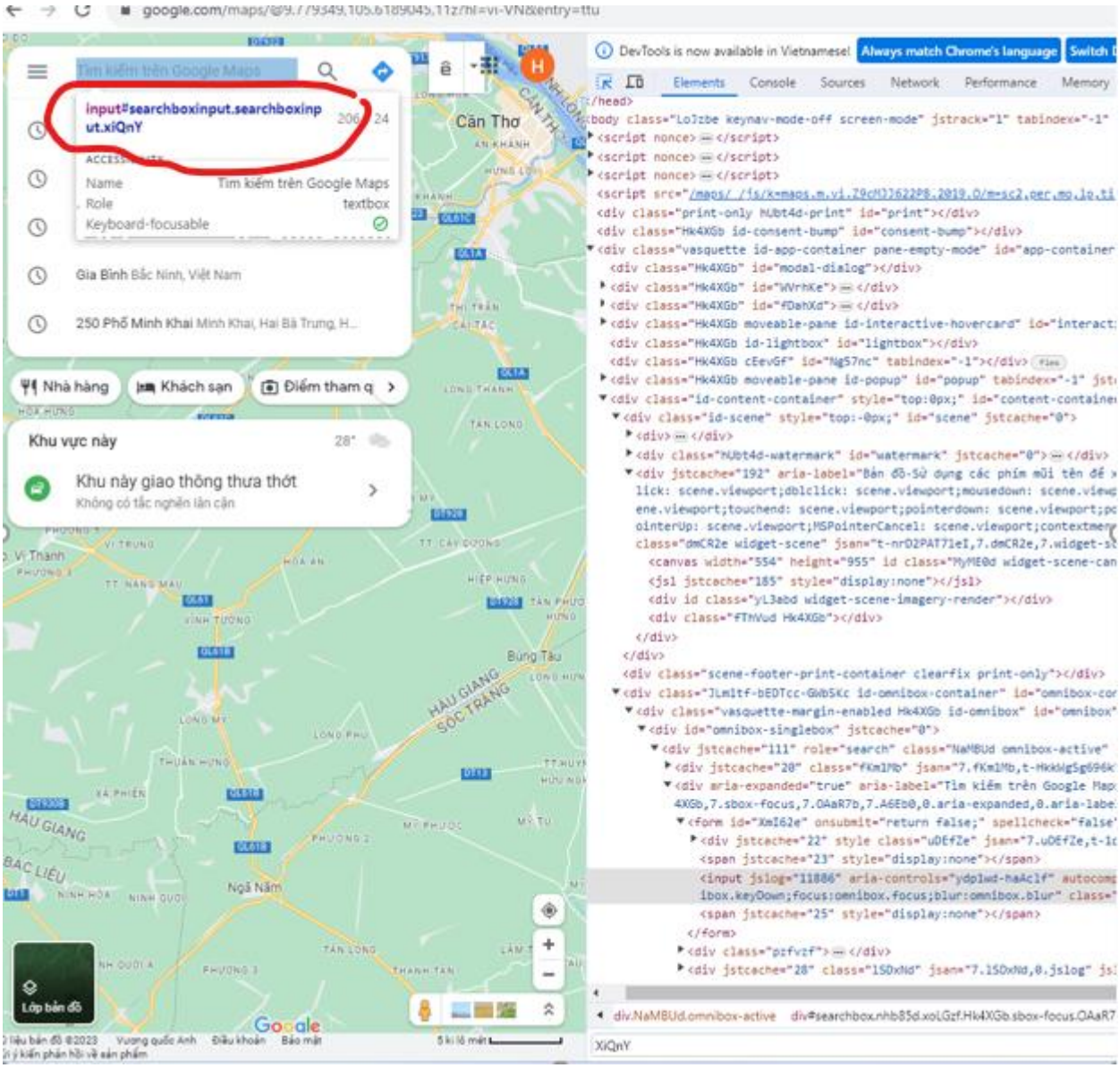
```python
import pandas as pd
import re
# đọc dữ liệu từ file excel
df = pd.read_excel("4g_output1_use_libRequest.xlsx")
print(df.info())
df1 = df[df['Address1'].isna()]
print(df1.info())

df2 = df[df['Address1'].notna()]
print(df2.info())



df1.to_excel('4g_output1_use_request_fail.xlsx', index=False)
df2.to_excel('4g_output1_use_request_win.xlsx', index=False)
```

4g_output1_use_request_fail sẽ đi qua tool Selenium

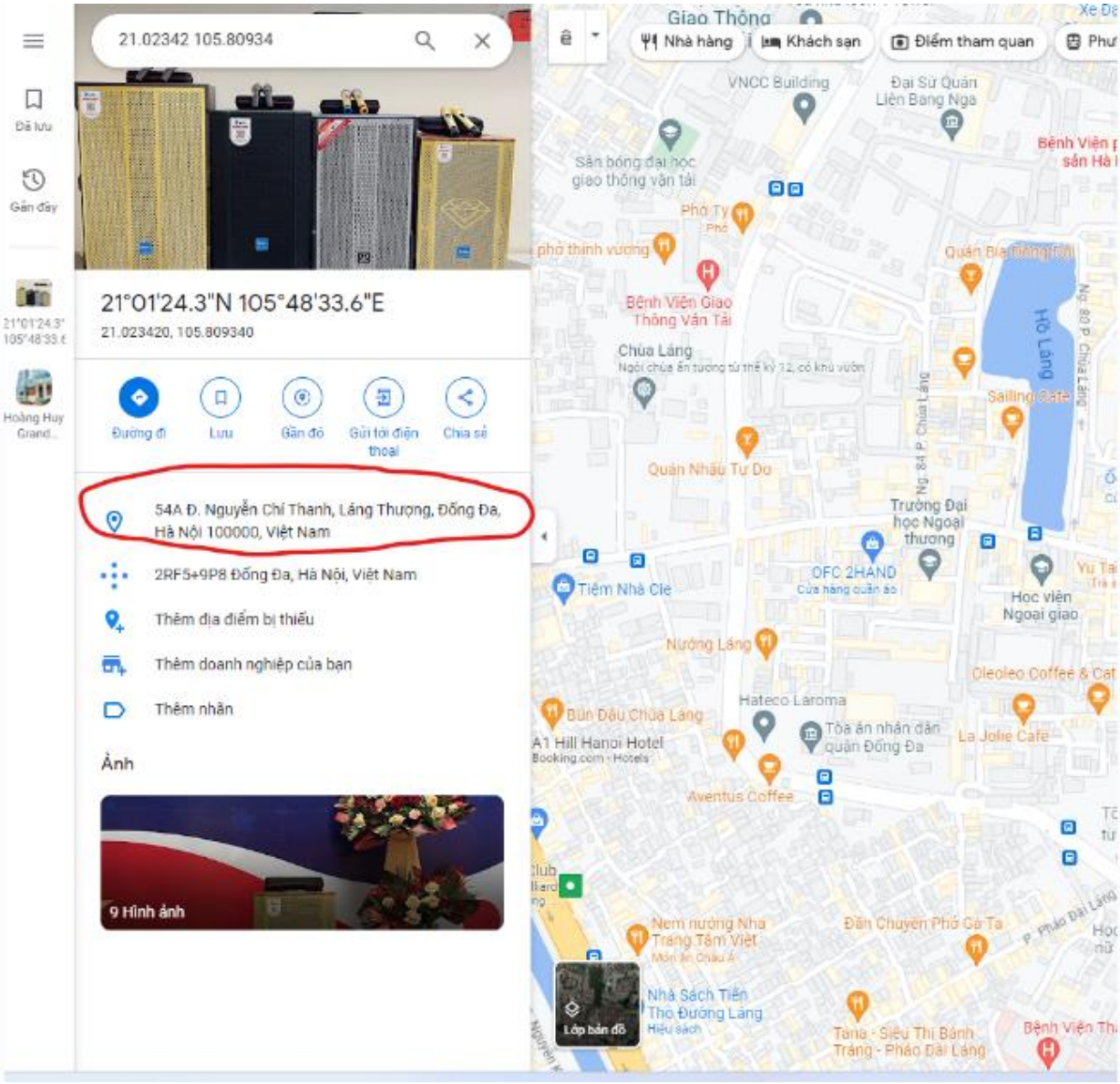
2. Sử dụng Selenium

```python
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
from bs4 import BeautifulSoup
import time
from selenium.webdriver.chrome.options import Options
import pandas as pd
import pickle


try:
    df = pd.read_excel("4g_output1_use_request_fail.xlsx")
    address2 = []
    latitude = df["LATITUDE"]
    longitude = df["LONGITUDE"]
    chrome_options = Options()
    # chạy nền ko hiển thị lên
    chrome_options.add_argument("--headless")
    driver = webdriver.Chrome(options= chrome_options)
    time.sleep(0.1)
    driver.get("https://www.google.com/maps")
    driver.implicitly_wait(20)
    for i in range (950,len(latitude)):
        print(i+1)
        if(i%50 == 49):
            time.sleep(1)
            driver = webdriver.Chrome(options= chrome_options)
            driver.get("https://www.google.com/maps")
            driver.implicitly_wait(30)

# Tìm ô tìm kiếm bằng ID
        search_box = driver.find_element(By.ID, "searchboxinput")
# Nhập tọa độ vào ô tìm kiếm
        search_box.send_keys(f"{latitude[i]}, {longitude[i]}")
# Cách 1
# Tìm nút tìm kiếm bằng ID và bấm vào nó
#search_button = driver.find_element(By.ID, "searchbox-searchbutton")
#search_button.click()
# Cách 2 gửi yêu cầu truy cập trực tiếp bằng phím ENTER
        driver.implicitly_wait(30)
        search_box.send_keys(Keys.ENTER)

# Chờ cho kết quả xuất hiện (thời gian chờ có thể cần điều chỉnh)
        driver.implicitly_wait(30)
        time.sleep(0.1)
# Thu thập thông tin về địa chỉ từ trang tìm kiếm
        place_address = driver.find_element(By.CSS_SELECTOR, "#QA0Szd > div > div > div.w6VYqd
```

Tìm ô searchbox để fill tọa độ vào



Sau đó thông tin trả về như sau



Để lấy được ở ô màu đỏ ta sử dụng css_selector tìm đúng vị trí cần lấy

Sau đó lại nhập tọa độ mới

```python
# Thu thập thông tin về địa chỉ từ trang tìm kiếm
        place_address = driver.find_element(By.CSS_SELECTOR, "#QA0Szd > div > div > div.w6VYqd
        driver.implicitly_wait(30)
        clear_box = driver.find_element(By.CSS_SELECTOR, "#searchbox > div.lSDxNd > button")
        clear_box.click()
        driver.implicitly_wait(30)
# In thông tin địa chỉ
        address2.append(place_address)
# Đóng trình duyệt
    driver.quit()

except Exception as e:
    # Xử lý lỗi (nếu cần)
    print("Xảy ra lỗi:", str(e))

finally:
    # Lưu danh sách thông tin thu thập được vào file pickle
    with open('address22.pkl', 'wb') as f:
        pickle.dump(address2, f)


print("____________________________________________________________")
print("Completed")
df["Address2"] = address2
# Tên tệp Excel là 'output.xlsx', index=False để không bao gồm cột index
df.to_excel('4g_output2_use_selenium.xlsx', index=False)
```

ở đoạn sau chúng ta sử dụng finally: vì chạy selenium hay bị đứt đường truyền . Đứt ở đâu thì lưu những thông tin đã thu thập được để khi chạy lại biết chạy từ chỗ nào

Đầu vào là file 4g_output1_use_request_fail

Đầu ra là file đã fill đầu đủ thông tin về địa chỉ (những có cả tòa nhà cả tên đường …)

Ta sẽ sử dụng filter để lọc các giá trị mong muốn

filter_selenium.py > ...

```python
import pandas as pd
import re
# đọc dữ liệu từ file excel
df = pd.read_excel("4g_output2_use_seleniumm.xlsx")
#Danh sách các từ bạn muốn tìm kiếm trong địa chỉ
desired_words = ['khu đô thị', 'tower', 'tòa', 'chung cư','building','vin', 'plaza', 'land', 'home','apartment', 'chung cu', 'căn hộ','center', 'nhà', 'KĐT',
                 'big c', 'văn phòng', 'bệnh viện', 'hotel','grand','khách sạn','trường','mall','bank','sun','garden','park','trung tâm','sky',
                 'pearl','công ty','ct','hh','win','house','town','holding','Riverside','department','mart','ngân hàng','cao ốc','Appartments','Khu thương mại',
                 'Tòa Nhà','Apartments','home','toa nha','celadon','block','gele','office','green','Tòa nhà' ]

# Tạo biểu thức chính quy với các từ bạn muốn tìm kiếm, kết hợp chúng bằng toán tử |
regex_pattern = '|'.join(map(re.escape, desired_words))
# Lọc các hàng trong đó cột 'Address' chứa ít nhất một từ trong danh sách
filtered_df_win = df[df['Address1'].str.contains(regex_pattern, case=False, na=False, regex=True)]
filtered_df_fail = df[~df['Address1'].str.contains(regex_pattern, case=False, na=False, regex=True)]
print(len(filtered_df_win))
print(len(df))
print(len(filtered_df_win)/len(df))
print(len(filtered_df_fail))

filtered_df_win.to_excel('4g_output2_use_seleniumm_win.xlsx', index=False)
filtered_df_fail.to_excel('4g_output2_use_seleniumm_fail.xlsx', index=False)
```

Sau khi Filter xong ta sẽ thu được 2 file .

1 file thỏa mãn yêu cầu lọc các các từ như chúng ta đã đặt ra : 4g_output2_use_seleniumm_win

1 file ko thỏa mãn : 4g_output2_use_seleniumm_fail

3.Ghép file

Ghép 1 file thỏa mãn ở phần sử dụng Open street map (4g_output1_use_request_win)và file thỏa mãn của phần selenium (4g_output2_use_seleniumm_win)

Ta được 1 file các tọa độ trả về đúng tên tòa nhà như yêu cầu (4g_win).ta sẽ biết được số lượng hàng trong file này là bao nhiêu để tí còn điền cột “status”

Ghép tiếp file vừa nhận được ở trên (4g_win). với file (4g_output2_use_seleniumm_fail) ta được file hoàn chỉnh (4g_complete). đủ số lượng bản ghi như file ban đầu (file 4g ban đầu được nhận là 3838 dòng, thì file này cũng đủ số lượng 3838 dòng như thế)

Tiếp theo thêm cột “status”

status.py > ...

```python
import pandas as pd

df = pd.read_excel("4g_win.xlsx")
length_4g_win = len(df)


df1 = pd.read_excel("4g_complete.xlsx")
df['status'] = 0  # Khởi tạo tất cả giá trị là 0
df['status'].iloc[:length_4g_win] = 1  # Gán giá trị 1 cho length hàng đầu tiên
print(df)
df.to_excel('4g_completeddd.xlsx', index=False)

```

Đầu vào file 4g_complete

Đầu ra là file 4g_completeddd đã được đánh trạng thái 0 1 (1 tức là ra được tên tòa nhà , 0 tức là chỉ ra được địa chỉ)

ở dòng thứ 4 chính là số dòng của file 4g_win