TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO MÔN XỬ LÝ DỮ LIỆU LỚN

BÀI BÁO CÁO GIỮA KỲ

Người hướng dẫn: **TS. Bùi Thanh Hùng**

Người thực hiện: **Trần Anh Vũ – 51800517**

Lớp: **18050201**

Khoá : **22**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM
TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG
KHOA CÔNG NGHỆ THÔNG TIN

BÁO CÁO MÔN XỬ LÝ DỮ LIỆU LỚN

BÀI BÁO CÁO GIỮA KỲ

Người hướng dẫn: **TS. Bùi Thanh Hùng**

Người thực hiện: **Trần Anh Vũ – 51800517**

Lớp: **18050201**

Khoá : **22**

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2021

LỜI CẢM ƠN

Để hoàn thành báo cáo này trước tiên em xin gửi đến cảm ơn thầy Bùi Thanh Hùng – người đã tận tình hướng dẫn, giúp đỡ em hoàn thành đố án cuối kì này lời cảm ơn sâu sắc nhất.

Với điều kiện thời gian cũng như kinh nghiệm còn chế của sinh viên, bài báo cáo này không thế tránh khỏi được những thiếu sót. Em rất mong nhận được sự chỉ bảo, đóng góp ý kiến của thầy để em có điều kiện bổ sung, nâng cao kiến thức của mình, phục vụ tốt hơn công việc thực tế sau này.

ĐỒ ÁN ĐƯỢC HOÀN THÀNH TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Tôi xin cam đoan đây là sản phẩm đồ án của riêng tôi / chúng tôi và được sự hướng dẫn của ThS. Bùi Thanh Hùng. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình. Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

TP. Hồ Chí Minh, ngày tháng năm

Tác giả

Trần Anh Vũ

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

Phần xác nhận của GV hướng dẫn

_____

_____

_____

_____

_____

_____

_____

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

Phần đánh giá của GV chấm bài

_____

_____

_____

_____

_____

_____

_____

Tp. Hồ Chí Minh, ngày tháng năm
(kí và ghi họ tên)

TÓM TẮT

Với sự phát triển của khoa học hiện đại rất nhanh chóng hiện nay dẫ đến tình trạng có nhiều dữ liệu khổng lồ cần phải xử lý và phân tích.

Do đó, qua bài giữa kỳ này em xin trình bày về hai phần mà trong đề giữa kỳ yêu cầu:

Phần 1 – Thu thập dữ liệu

Phần 2 – Khai phá dữ liệu

MỤC LỤC

DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT

CÁC KÝ HIỆU

CÁC CHỮ VIẾT TẮT

df – DataFrame

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

DANH MỤC HÌNH

DANH MỤC BẢNG

PHẦN 1 – THU THẬP DỮ LIỆU

1.1 Bạn hãy viết code cào dữ liệu từ trang web trên, lưu kết quả vào 1 file tương ứng (kq.txt) và mô tả ngắn gọn về cấu trúc của trang Web trên?

- Do phần 1 yêu cầu sử cào dữ liệu từ web với ít nhất 40 dòng nên em đã chọn selenium để thực hiện việc cào dữ liệu tự động ở các trang:
- Đầu tiên là tải driver trên trang chủ cảu selenium(cần tải driver phù hợp với trình duyệt muốn sử dụng), ở đây em sẽ dùng chrome version 100 để làm bài.
- Sau đó import các thư viện để sử dụng.

```
#import thư viện selenium
from selenium import webdriver
from selenium.webdriver.chrome.service import Service
from webdriver_manager.chrome import ChromeDriverManager
from selenium.webdriver.common.by import By
##
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC
###
MAX_PAGE_NUM = 5 # số trang muốn lấy
##
import csv
##
import pandas as pd
import datetime as dt
import numpy as np
from dateutil.relativedelta import relativedelta
```

Ảnh 1 Khai báo thư viện

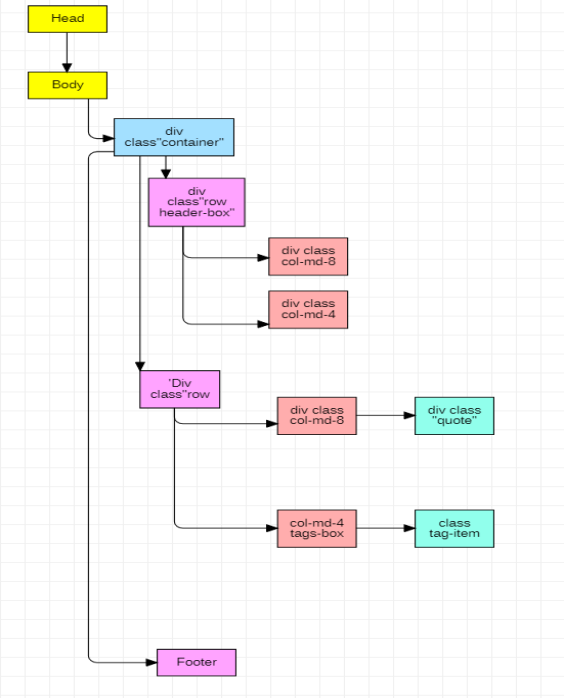
- Tiếp theo em sẽ khai báo driver để chạy selenium và sử dụng phương thức get để chỉ định trang web cần cào:
-
- Tiếp theo em sẽ thực hiện code về giai đoạn lấy tất cả nội dụng của ý 1.1 và lưu vào file ‘kq.txt’.

```python
#Câu 1.1
for i in range(MAX_PAGE_NUM):
    #Xác định những phần tử muốn lấy
    E_AllPage = driver.find_element(By.CLASS_NAME, "container").text
    t = str(E_AllPage)
    file = open("./kq.txt","a+")#tên file chứa dữ liệu của trang đầu của web, ghi đè với chế độ a+
    file.writelines(t)

    print('Trang',i+1,': Đã lưu thành công!')
    file.close()

    file = open("./kq.txt","a+")#tên file chứa dữ liệu của trang đầu của web
    file.writelines('\n\n nextpage----------------------------------------------------------------------------------------------------\r
    next = driver.find_element(By.CSS_SELECTOR, ".next [href]").click()# xác định hrè để lấy trang kế tiếp
file.close()
driver.get("http://quotes.toscrape.com/")
```

```
Trang 1 : Đã lưu thành công!
Trang 2 : Đã lưu thành công!
Trang 3 : Đã lưu thành công!
Trang 4 : Đã lưu thành công!
Trang 5 : Đã lưu thành công!
```

Ảnh 2 Code để lấy toàn phần của web

- Mô tả về cấu trúc của trang web(em sẽ mô tả về cấu trúc web bằng sơ đồ sau):



Ảnh 3 Cấu trúc sơ bộ của trang web https://quotes.toscrape.com/

- Mô tả về cấu trúc trên trang web: 1. Web chia ra các thẻ và tên class như sau:
    - Body:
        - Thẻ div class="container"

- Thẻ div class="row header-box" chứa tên của trang web(Quotes to Scrape) và vị trí để login tài khoản
- Thẻ row chứa hai thẻ có tên class="col-md-8"(chứa phần lớn nội dung chính của trang như các câu nói, tên tác giả, thông tin của tác giả và các tags(liên quan)) và class="col-md-4 tags-box"(chứa những tag được phổ biến trong top 10(Top Ten tags))
- Thẻ 'ul' chia trang trước và trang sau.
- Cuối cùng là thẻ div class= "Footer"(chứa các thông tin cơ bản về trang web)

1.2 Với dữ liệu bạn vừa cào về, bạn hãy thực hiện các yêu cầu sau:

- Với câu 1.2 ý 1:
- Em thực hiện như sau em sẽ cào 5 trang của web và in ra màn hình những dòng có class theo yêu cầu.

```
#Cau1.2a
print('Kết quả câu 1.2a')
for i in range(MAX_PAGE_NUM):
    #Xác định những phần tử muốn lấy
    print('\n\nPage',i+1,'\n')
    #chờ đợi cho đến khi hiển thị tất cả element của class'quote'
    result = WebDriverWait(driver, 20).until(EC.visibility_of_all_elements_located((By.CLASS_NAME, "quote")))

    for print_result in result:
        print(print_result.text)
        print("\n")

    next = driver.find_element(By.CSS_SELECTOR, ".next [href]").click()# xác định href để lấy trang kế tiếp
driver.get("http://quotes.toscrape.com/")
```

Ảnh 4 Câu 1.2 ý 1

```
Kết quả câu 1.2a


Page 1

“The world as we have created it is a process of our thinking. It cannot be changed without changing our thinking.”
by Albert Einstein (about)
Tags: change deep-thoughts thinking world


“It is our choices, Harry, that show what we truly are, far more than our abilities.”
by J.K. Rowling (about)
Tags: abilities choices


“There are only two ways to live your life. One is as though nothing is a miracle. The other is as though everything is a mir
acle.”
by Albert Einstein (about)
Tags: inspirational life live miracle miracles
```

Ảnh 5 Kết quả của câu 1.2 ý 1

- Tiếp đến là phần 1.2 ý 2 đề sẽ yêu cầu lấy dữ liệu ở các nhãn <small> và có class"author", và sau khi xem xét thì kết quả là sẽ phải lấy tên của tác giả.

```
 #Câu 1.2b: Sau khi in ra ta thấy trong nhãn<small> đều có chung các class="author" giống nhau nên sẽ dùng class name để
print('Kết quả câu 1.2a: ')
for i in range(MAX_PAGE_NUM):
    #Xác định những phần tử muốn lấy
    print('\n\nPage',i+1,'\n')
    #chờ đợi cho đến khi hiển thị tất cả element của class'quote'
    result_author = driver.find_elements(By.CLASS_NAME, "author")

    for print_resultAuthor in result_author:
        print('AuthorName_Page-',i+1,': ',print_resultAuthor.text)

    next = driver.find_element(By.CSS_SELECTOR, ".next [href]").click()# xác định href để lấy trang kế tiếp
driver.get("http://quotes.toscrape.com/")
```

```
Page 1

AuthorName_Page- 1 :  Albert Einstein
AuthorName_Page- 1 :  J.K. Rowling
AuthorName_Page- 1 :  Albert Einstein
AuthorName_Page- 1 :  Jane Austen
AuthorName_Page- 1 :  Marilyn Monroe
AuthorName_Page- 1 :  Albert Einstein
AuthorName_Page- 1 :  André Gide
AuthorName_Page- 1 :  Thomas A. Edison
AuthorName_Page- 1 :  Eleanor Roosevelt
AuthorName_Page- 1 :  Steve Martin
```

Ảnh 6 Code và kết quả của câu 1.2 ý 2

- Tiếp theo để thực hiện yêu cầu viết hàm tacgiaLink() để in ra kết quả bao gồm Tên tác giả, Đường link của tác giả, Ngày tháng năm sinh, câu nói nổi tiếng của tác giả. Em sẽ làm như sau:
    - Đầu tiên em sẽ viết một đoạn code để lấy dữ liệu ngày sinh trước vì nếu muốn lấy dữ liệu ngày sinh phải truy cập vào một link khách nên em đã lấy trước và lưu vào biến: data0, data1, data2, data3, data4(tương úng từ trang 1 đến 5).

```
nameAuthor = driver.find_elements(By.CLASS_NAME, "author")
num = len(nameAuthor)
data0 = []
data1 = []
data2 = []
data3 = []
data4 = []
data_born = []
for i in range(MAX_PAGE_NUM):
    print("\npage:",i+1,"\n")
    for j in range(num):
        if i == 0:
            click_link = driver.find_elements(By.XPATH, '//a[text() = "(about)"]')[j].click()
            born = driver.find_element(By.XPATH, '//span[@class="author-born-date"]')
            print(born.text)
            data0.append(born.text)
            data_born.append(born.text)

            driver.back()
        elif i == 1:
            click_link = driver.find_elements(By.XPATH, '//a[text() = "(about)"]')[j].click()
            born = driver.find_element(By.XPATH, '//span[@class="author-born-date"]')
            print(born.text)
            data1.append(born.text)
            data_born.append(born.text)
            driver.back()
        elif i == 2:
            click_link = driver.find_elements(By.XPATH, '//a[text() = "(about)"]')[j].click()
            born = driver.find_element(By.XPATH, '//span[@class="author-born-date"]')
            print(born.text)
            data2.append(born.text)
            data_born.append(born.text)
            driver.back()
        elif i == 3:
            click_link = driver.find_elements(By.XPATH, '//a[text() = "(about)"]')[j].click()
            born = driver.find_element(By.XPATH, '//span[@class="author-born-date"]')
            print(born.text)
            data3.append(born.text)
            data_born.append(born.text)
            driver.back()
        elif i == 4:
            click_link = driver.find_elements(By.XPATH, '//a[text() = "(about)"]')[j].click()
            born = driver.find_element(By.XPATH, '//span[@class="author-born-date"]')
            print(born.text)
            data4.append(born.text)
            data_born.append(born.text)
            driver.back()
        else:
            click_link = driver.find_elements(By.XPATH, '//a[text() = "(about)"]')[j].click()
            born = driver.find_element(By.XPATH, '//span[@class="author-born-date"]')
#             print(born.text)
            data_born.append(born.text)
            driver.back()
    next = driver.find_element(By.CSS_SELECTOR, ".next [href]").click()# xác định href để lấy trang kế tiếp
driver.get("http://quotes.toscrape.com/")
print('done....')
```

Ảnh 7 Code để lấy ngày sinh của tác giả

- Sau đó với những dữ liệu cần lấy còn lại như(tên tác giả, đường link, câu nói) em sẽ lấy theo các phương thức như sau: tên tác giả(class"author"), đường link(xpath), câu nói(xpath). Và sau đó in ra:

Error! Bookmark not defined.

```
def tacgiaLink():
    print('Kết quả câu 1.2c: ')
    for i in range(MAX_PAGE_NUM):
        print('\n\nPage',i+1,'\n')

        nameAuthor = driver.find_elements(By.CLASS_NAME, "author")
        hrefAbout = driver.find_elements(By.XPATH, '//a[text() = "(about)"]')
        famous_Quote = driver.find_elements(By.XPATH, '//span[@class="text"]')
#       bornHref = driver.find_elements(By.XPATH, '//a[text() = "(about)"]')
#       bornAuthor = driver.find_element(By.CLASS_NAME,"author-born-date")

        num = len(nameAuthor)
        for j in range(num):
            if i == 0:
                print( "Tên tác giả: " + nameAuthor[j].text + "\nĐường link của tác giả: " + hrefAbout[j].get_attribute('href') +
            elif i == 1:
                print( "Tên tác giả: " + nameAuthor[j].text + "\nĐường link của tác giả: " + hrefAbout[j].get_attribute('href') +
            elif i == 2:
                print( "Tên tác giả: " + nameAuthor[j].text + "\nĐường link của tác giả: " + hrefAbout[j].get_attribute('href') +
            elif i == 3:
                print( "Tên tác giả: " + nameAuthor[j].text + "\nĐường link của tác giả: " + hrefAbout[j].get_attribute('href') +
            elif i == 4:
                print( "Tên tác giả: " + nameAuthor[j].text + "\nĐường link của tác giả: " + hrefAbout[j].get_attribute('href') +

        next = driver.find_element(By.CSS_SELECTOR, ".next [href]").click()# xác định href để lấy trang kế tiếp
    driver.get("http://quotes.toscrape.com/")
    print("done")
```

Ảnh 8 Hàm tacgiaink()

```
: tacgiaLink()
Kết quả câu 1.2c:


Page 1

Tên tác giả: Albert Einstein
Đường link của tác giả: http://quotes.toscrape.com/author/Albert-Einstein
Ngày tháng năm sinh: March 14, 1879
Quote: “The world as we have created it is a process of our thinking. It cannot be changed without changing our thinking.”

Tên tác giả: J.K. Rowling
Đường link của tác giả: http://quotes.toscrape.com/author/J-K-Rowling
Ngày tháng năm sinh: July 31, 1965
Quote: “It is our choices, Harry, that show what we truly are, far more than our abilities.”

Tên tác giả: Albert Einstein
Đường link của tác giả: http://quotes.toscrape.com/author/Albert-Einstein
Ngày tháng năm sinh: March 14, 1879
Quote: “There are only two ways to live your life. One is as though nothing is a miracle. The other is as though everything
```

Ảnh 9 Kết quả hàm tacgiaLink()

- Và yêu cầu cuối cùng là lưu kết quả vào file Quote.csv:

```
# + 1.2d. Hãy lưu kết quả ở câu c vào file Quote.csv tương ứng, với mỗi tác giả là 1
# dòng dữ liệu. Bạn được yêu cầu thu thập ít nhất 40 câu nói nổi tiếng từ trang
# web trên một cách tự động theo code của các ý trên? (5 điểm)
for i in range(MAX_PAGE_NUM):
    print('\nPage',i+1,'\n: đã ghi vào file csv!')

    nameAuthor = driver.find_elements(By.CLASS_NAME, "author")
    hrefAbout = driver.find_elements(By.XPATH, '//a[text() = "(about)"]')
    famous_Quote = driver.find_elements(By.XPATH, '//span[@class="text"]')
    header= ['Tacgia', 'Link', 'Namsinh','Quote']
    num = len(nameAuthor)
    with open('Quote.csv', 'a', encoding='utf-8-sig') as f:
        writer = csv.writer(f)
        writer.writerow(header)
        for j in range(num):
            if i == 0:
                writer.writerow([nameAuthor[j].text, hrefAbout[j].get_attribute('href'), data_born[j], famous_Quote[j].text])
                data = ([nameAuthor[j].text, hrefAbout[j].get_attribute('href'), data0[j], famous_Quote[j].text])
                print(data)
            elif i == 1:
                writer.writerow([nameAuthor[j].text, hrefAbout[j].get_attribute('href'), data_born[j], famous_Quote[j].text])
                data = ([nameAuthor[j].text, hrefAbout[j].get_attribute('href'), data1[j], famous_Quote[j].text])
                print(data)
            elif i == 2:
                writer.writerow([nameAuthor[j].text, hrefAbout[j].get_attribute('href'), data_born[j], famous_Quote[j].text])
                data = ([nameAuthor[j].text, hrefAbout[j].get_attribute('href'), data2[j], famous_Quote[j].text])
                print(data)
            elif i == 3:
                writer.writerow([nameAuthor[j].text, hrefAbout[j].get_attribute('href'), data_born[j], famous_Quote[j].text])
                data = ([nameAuthor[j].text, hrefAbout[j].get_attribute('href'), data3[j], famous_Quote[j].text])
                print(data)
            elif i == 4:
                writer.writerow([nameAuthor[j].text, hrefAbout[j].get_attribute('href'), data_born[j], famous_Quote[j].text])
                data = ([nameAuthor[j].text, hrefAbout[j].get_attribute('href'), data4[j], famous_Quote[j].text])
                print(data)
    next = driver.find_element(By.CSS_SELECTOR, ".next [href]").click()# xác định href để lấy trang kế tiếp
driver.get("http://quotes.toscrape.com/")
```

Ảnh 10 code lưu dữ liệu vào file csv

PHẦN 2 – KHAI PHÁ DỮ LIỆU

2.1 Xử lý dữ liệu- Data Imputation

2.1.1 Phần giới thiệu:

- Với những dữ liệu khi được lấy về thì không phải dữ liệu nào cũng được định dạng sẵn hay có đầy đủ các ô dữ liệu cho nên cần phải vận dụng các kỹ thuật xử lý dữ liệu để xử lý những vấn đề như missing data.

2.1.2 Cách tiếp cận

- Sau khi thực hiện yêu cầu 2.1 ý 1với dữ liệu em thu về từ web ở phần 1 sẽ không có ô bị thiếu như đề cần tìm hiểu nên em có lấy 1 ví dụ đơn giản về missing data và đề xuất cách xử lý nó.
- Đầu tiên em sẽ tạo dataframe với hai dòng như ảnh dưới và thực hiện xử lý nó:

```
#2.1
- Giả định trường hợp dữ liệu trường ngày sinh bị thiếu nhiều dòng, em đề xuất điền bằng hàm `filla()` nó sẽ giúp điền vào những vị trí thiếu trong dataFrame một cách có chỉ định.
```

```
34]: #vd2.1
df_vd1 = pd.DataFrame({'A':[np.nan, 2, np.nan, 0],'B':[3, 4, np.nan, 1]})
print('Dữ liệu bị missing\n',df_vd1)
```

```
Dữ liệu bị missing
     A    B
0  NaN  3.0
1  2.0  4.0
2  NaN  NaN
3  0.0  1.0
```

Ảnh 11 df có missingvalue

- Với trường hợp này em sẽ đề xuất dùng hàm fillna() để có thể thay thế các giá trị missing.

```
df_vd1=df_vd1.fillna(0)
print('Dữ liệu đã được thay thế \n',df_vd1)
```

```
Dữ liệu đã được thay thế
      A    B
0  0.0  3.0
1  2.0  4.0
2  0.0  0.0
3  0.0  1.0
```

Ảnh 12 dữ liệu missing được thay bằng số 0

- Tiếp theo ở ý 2 để thực hiện việc thêm trường tuổi và đề xuất cách điền tuổi của các tác giả
- Đầu tiên em sẽ đề xuất phương án điền số tuổi của các tác giả theo cách em làm: đầu tiên em sẽ chuyển dữ liệu của trường ‘Namsinh’ về int64(dạng datetime) trước
- Tiếp theo, em sẽ dùng ‘relativedelta’ để thực hiện lấy ngày hiện tại để trừ cho trường năm sinh để lấy được tuổi của tác giả cho đến hiện tại. Và thực hiện thêm cột tuổi vào dataframe.

```
#tính số Tuoi của tác giả cho đến hiện tại
df['Namsinh']
s = [relativedelta(pd.to_datetime('now'), d).years for d in df['Namsinh']]
s=list(s)
```

```
#tạo thêm cột tuổi vào df
#Cách thêm tuổi vào df là em sẽ tạo thêm một cột mới
df['Tuoi'] = s
df.update(df)#update dữ liệu mới khi được thêm cột
df
```

Ảnh 13 code điền tuổi và thêm trường ‘Tuoi’ của tác giả vào df

2.1.3 Đánh giá

- Bước xử lý dữ liêu này rất quan trọng và là bước đầu tiên để xử lý dữ liệu với mục đích làm sạch dữ liệu mới được thu về. Đây là một bước rất quan trọng trong việc xử lý dữ liệu lớn.

2.2 Khám phá dữ liệu- Data Exploration

2.2.1 Phần giới thiệu

- Khám phá dữ liệu là bước mà khi đã xử lý dữ liệu xong, sẽ tiến hành dùng dữ liệu đó để tạo ra các phân tích ban đầu về dữ liệu để phục vụ cho các bước phân tích chuyên sâu về sau. Nhìn chung nó sẽ giúp người thu thập hiểu rõ hơn về dữ liệu mới được thu thập.

2.2.2 Cách tiếp cận

- Để tiếp cận được phương pháp này em sẽ tiến hành các yêu cầu của đề bài:
  - Thống kê về tác giả và câu nói nổi tiếng có trong bộ dữ liệu: Để thực hiện yêu cầu này em sẽ dùng hàm groupby và thống kê tác giả và những câu nói trong dữ liệu.

```
#Cau1.2a: #Thống kê về tác giả và câu nói nổi tiếng có trong bộ dữ liệu,
df2_1a_sum = df.groupby("Tacgia")["Quote"].sum()
print("Tên tác giả và các câu nói: \n", df2_1a_sum)
```

```
Tên tác giả và các câu nói:
```

Ảnh 14 Thống kê về tác giả và các câu nói

```
df2_1a_count = df.groupby("Tacgia")["Quote"].count()

print("Tổng số câu nói của từng các giả:",df2_1a_count)
```

```
Tổng số câu nói của từng các giả: Tacgia
Albert Einstein          8
Allen Saunders           1
André Gide               1
Bob Marley               3
C.S. Lewis               1
```

Ảnh 15 tác giả và số câu nói

  - Thống kê về năm sinh và độ tuổi của các tác giả: yêu cầu này em sẽ gom nhóm những tác giác có cùng độ tuổi với nhau:

```
#cau1.2b Thống kê về năm sinh và độ tuổi của các tác giả,
df2_1b = df.groupby("Tuoi")["Tacgia"].sum()
print("Các tác giả có cùng độ tuổi: \n",df2_1b,end='\n')
```

```
Các tác giả có cùng độ tuổi:
 Tuoi
56     J.K. RowlingJ.K. RowlingRalph Waldo EmersonJ.K. RowlingC.S. Lewis
76     Steve MartinAllen SaundersBob MarleyGeorge EliotJames Baldwin
95     Marilyn MonroeDr. SeussJim HensonJ.K. RowlingAlbert Einstein
137    Eleanor RooseveltMark TwainAlbert EinsteinJorge Luis BorgesJ.K. Rowling
143    Albert EinsteinAlbert EinsteinAlbert EinsteinMarilyn MonroeAlbert EinsteinDouglas Adam
sDr. SeussBob MarleyCharles M. SchulzGeorge R.R. MartinMarilyn MonroeMarilyn Monroe
152    André GideElie WieselAlbert EinsteinWilliam NicholsonMarilyn Monroe
175    Thomas A. EdisonFriedrich NietzscheJ.K. RowlingAlbert EinsteinMartin Luther King Jr.
246    Jane AustenBob MarleyGarrison KeillorMother TeresaMarilyn Monroe
Name: Tacgia, dtype: object
```

Ảnh 16 Các tác giả có cùng độ tuổi

- Thống kê về năm sinh và độ tuổi:

```
]: #Thống kê về năm sinh và độ tuổi
   df.describe(datetime_is_numeric=True)
```

]:

	Namsinh	Tuoi
count	50	50.000000
mean	1885-04-17 14:24:00	136.600000
min	1775-12-16 00:00:00	56.000000
25%	1869-11-22 00:00:00	95.000000
50%	1879-03-14 00:00:00	143.000000
75%	1926-06-01 00:00:00	152.000000
max	1965-07-31 00:00:00	246.000000
std	NaN	51.212243

Ảnh 17 Thống kê về năm sinh và độ tuổi

- Thống kê về các câu nói nổi tiếng như: câu dài nhất, ngắn nhất, số từ:

```
]: #Cau 2_1c#Thống kê về các câu nói nổi tiếng như: câu dài nhất, ngắn nhất
   count = df['Quote'].str.split().str.len()
   count.index = "Câu "+ df['Quote']+ ' Có:'
   count.sort_index(inplace=True)
   c_max = count.max()
   c_min = count.min()
   c_mean = count.mean()
   print("Câu dài nhất có" , c_max , "Từ")
   print("Câu ngắn nhất có" , c_min , "Từ")
   print("Trung bình 1 câu có " , c_mean , "Từ")
```

```
Câu dài nhất có 201 Từ
Câu ngắn nhất có 7 Từ
Trung bình 1 câu có  24.32 Từ
```

Ảnh 18 Khảo sát về độ dài các câu nói

- Số lượng những tác giả khác nhau: với dữ liệu thu được thì các tác giả sẽ có nhiều câu nói, khảo sát này sẽ giúp chúng ta nhận thấy số lượng tác giả trong 50 câu nói thu được.

```
]: #Số lượng những tác giả khác nhau
   df2_1e=df.Tacgia.nunique()
   print(df2_1e)

   28
```

Ảnh 19 Số lượng tác giả trong df

2.2.3 Đánh giá

- Nhìn chung thì bước này sẽ giúp người xử lý hiểu hơn về dữ liệu mà thu được sẽ giúp định hướng mục đích dùng dữ liệu vào các ứng dụng khác nhau. Bằng việc vận dụng thư viện pandas sẽ giúp dễ dàng hoàn thành hơn.
- Giúp giảm tải những dữ liệu cần xử lý bằng cách biện pháp gom nhóm.

TÀI LIỆU THAM KHẢO

[1] https://www.selenium.dev/documentation/

[2] https://www.analyticsvidhya.com/blog/2020/08/web-scraping-selenium-with-python/

[3] https://bigdatauni.com/tin-tuc/cac-phuong-phap-imputation-don-gian-cho-missing-values.html

[4] https://pandas.pydata.org/docs/reference/frame.html

[5] https://www.sisense.com/glossary/data-exploration/