**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM**

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**BÁO CÁO GIỮA KÌ**

**MÔN XỬ LÝ DỮ LIỆU LỚN**

**THU NHẬP VÀ KHAI PHÁ BỘ DỮ LIỆU**

*Người hướng dẫn*: Thầy BÙI THANH HÙNG

*Người thực hiện*: Nguyễn Hoàng Quang Nhật\_51800220

Khoá : 22

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022

**TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM**

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**

****

**BÁO CÁO GIỮA KÌ**

**MÔN XỬ LÍ DỮ LIỆU LỚN**

**THU NHẬP VÀ KHAI PHÁ BỘ DỮ LIỆU**

*Người hướng dẫn*: Thầy BÙI THANH HÙNG

*Người thực hiện*: Nguyễn Hoàng Quang Nhật\_51800220

Khoá : 22

THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022

LỜI CẢM ƠN

Em xin cảm ơn đến thầy Hùng vì đã giúp đỡ chúng em trong quá trình làm bài giữa kỳ môn xử lý dữ liệu lớn. Nếu không có thầy em nghĩ mình sẽ không thể hoàn thành bài tập này một cách tốt nhất được. Em xin được trân trọng gửi lời cảm ơn đến thầy Hùng.

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Nhóm em xin cam đoan đồ án của mình là sản phẩm riêng của bản thân và dưới sự hướng dẫn của thầy Bùi Thanh Hùng. Kết quả nghiên cứu này là hoàn toàn trung thực. Những tư liệu phục vụ cho đồ án đều ghi nguồn tham khảo rõ ràng ở mục tài liệu tham khảo.

Nếu có phát hiện gian dối trong đồ án môn học này, em xin hoàn toàn chịu trách nhiệm.

*TP. Hồ Chí Minh, ngày 4 tháng 4 năm 2022*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

Nguyễn Hoàng Quang Nhật

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

Những người nổi tiếng trên thế giới đã đóng góp rất nhiều thành tựu cho sự thay đổi của thế giới,những câu nói của họ có htế khiến cho người khác phải thay đổi tư duy,cách sống của rất nhiều người khác.

Tróng báo cáo này chúng ta sẽ tìm hiểu về cách thu thập và khái phá trên bộ dữ liệu này

MỤC LỤC

[LỜI CẢM ƠN 3](#_Toc100000411)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN 4](#_Toc100000412)

[TÓM TẮT 6](#_Toc100000413)

[MỤC LỤC 7](#_Toc100000414)

[DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT 7](#_Toc100000415)

[CHƯƠNG I – THU NHẬP DỮ LIỆU 8](#_Toc100000416)

[1.1 Viết code cào dữ liệu từ trang web Quotes to Scrape: 8](#_Toc100000417)

[1.2 Với dữ liệu bạn vừa cào về, bạn hãy thực hiện các yêu cầu sau: 9](#_Toc100000418)

[CHƯƠNG II – KHAI PHÁ DỮ LIỆU 13](#_Toc100000419)

[2.1 Xử lý dữ liệu - Data Imputation: 13](#_Toc100000420)

[Đề xuất cách điền cho một số trường ngày sinh chưa có: 13](#_Toc100000421)

[Đề xuất cách điền thêm trường Tuoi(Tuổi): 14](#_Toc100000422)

[2.3 Tiền xử lý dữ liệu - Data Preprocessing: 16](#_Toc100000423)

[2.4 Khám phá dữ liệu - Data Exploration: 17](#_Toc100000424)

[Thống kê số câu nói của các tác giả: 18](#_Toc100000425)

[Thống kê tuổi của tác giả: 19](#_Toc100000426)

[Thống kê năm sinh của tác giả: 20](#_Toc100000427)

[Thống kê về câu dài nhất, ngắn nhất, số từ: 21](#_Toc100000428)

[2.5 Trích xuất đặc trưng - Feature Extraction: 22](#_Toc100000429)

[CHƯƠNG III – TỔNG KẾT 23](#_Toc100000430)

[3.1 Cài đặt các thư viện yêu cầu: 23](#_Toc100000431)

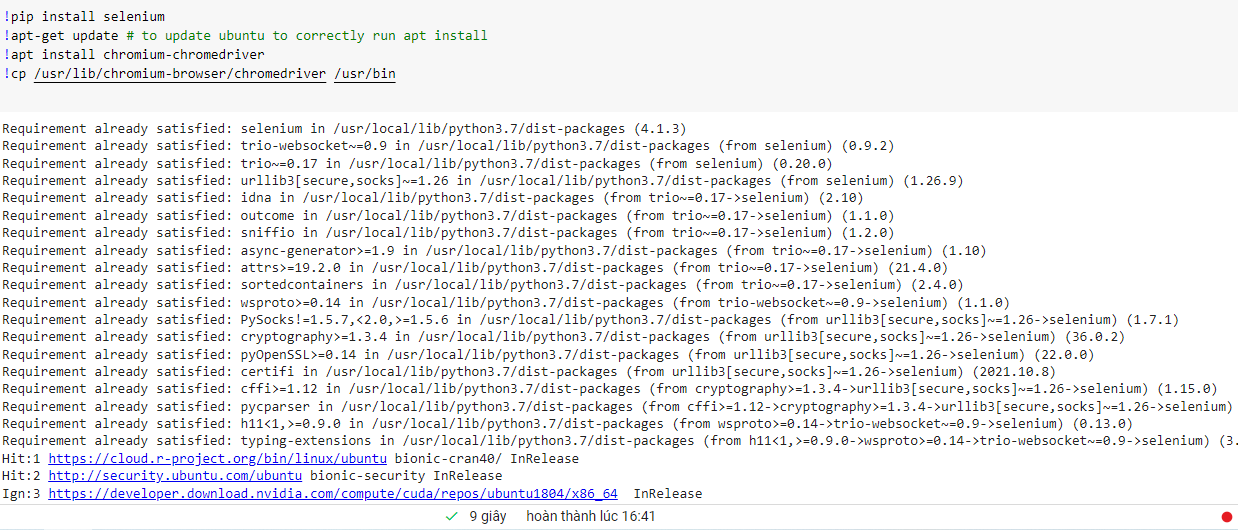
[3.2 Tài liệu tham khảo: 24](#_Toc100000432)

DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT

CHƯƠNG I – THU NHẬP DỮ LIỆU

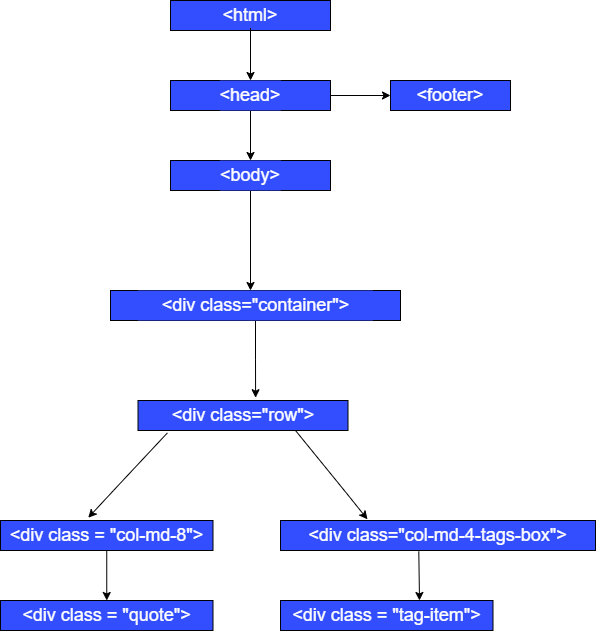
1.1 Viết code cào dữ liệu từ trang web Quotes to Scrape:

Theo đề bài chúng ta sẽ phải crawl dữ liệu câu nói và các thông tin tác giả từ nguồn [Quotes to Scrape](http://quotes.toscrape.com/) cho nên chúng ta sẽ sử dụng 2 module hỗ trợ crawl dữ liệu rất mạnh mẽ đó là urllib và BeautifulSoup. Tuy nhiên do chạy trên colab chúng ta sẽ cần phải cài đặt một số thứ để chạy selenium trên colab.



Hình 1.1.1 thêm các thư viện vào để xử lý





*Hình 1.1.1b Mô tả cấu trúc trang web*

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

Tiếp đến đó là viết các hàm để crawl và lưu dữ liệu theo yêu cầu

1. Hãy đọc tất cả các thẻ html (div) với lớp là "quote" và lưu nó trong biến

'result’, hiển thị giá trị biến 'result’ ra màn hình?

1. Hãy tìm trong biến 'result’ vừa rồi các dữ liệu có chứa nhãn "small" với

class là "author" và in kết quả ra màn hình?



*Hình 1.1.2 Đọc các thẻ div có class là quote và tìm trong biến result có thẻ small class là author cùng với đó là tìm tên tác giả sau đó in kết quả*

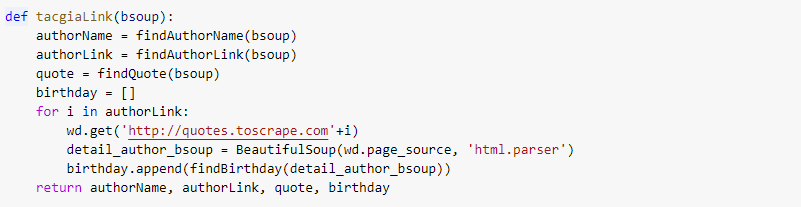
1. Hãy viết hàm tacgiaLink() để lấy nội dung của mỗi tác giả. Với mỗi tác giả

in ra màn hình các nội dung

* Tên tác giả
* Đường link tác giả
* Ngày tháng năm sinh
* Và câu nói nổi tiếng của tác giả

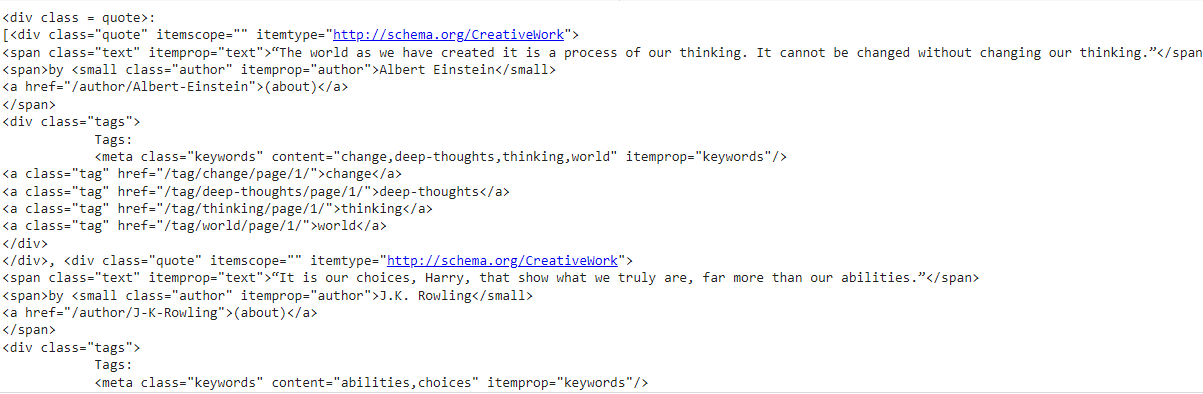


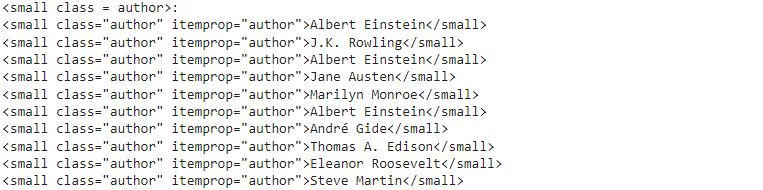
*Hình 1.1.3 & 1.1.4 viết hàm tacgialink và dùng các hàm để lấy link tác giả, ngày tháng năm sinh và câu nói nổi tiếng của tác giả*

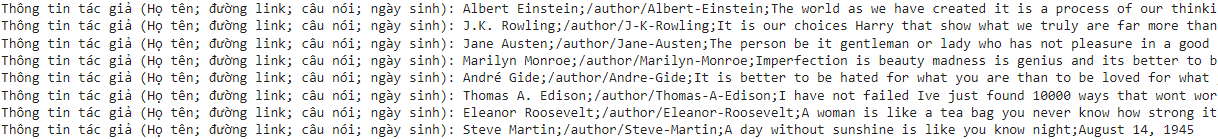




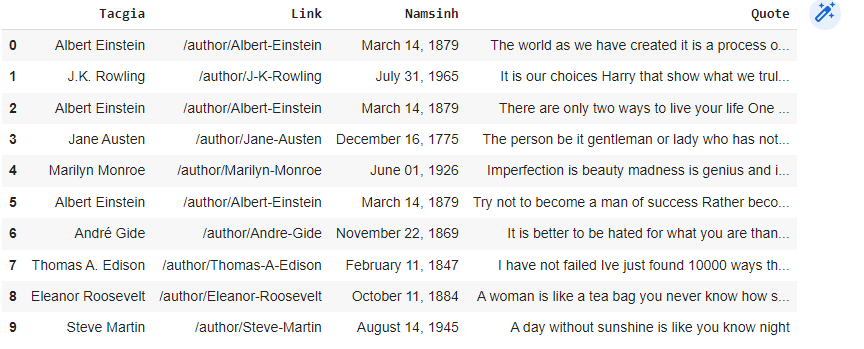
*Hình 1.1.5 Sau khi viết các hàm bắt đầu in ra thông tin ,kết quả lưu vào dataFrame*







*Hình 1.1.6 kết quả sau khi viết các hàm*



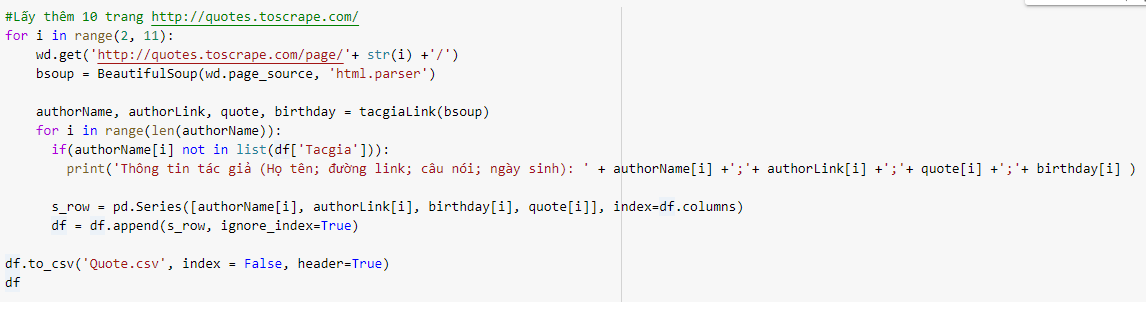
*Hình 1.1.7 bảng dữ liệu hoàn chỉnh*

1. Hãy lưu kết quả ở câu c vào file Quote.csv tương ứng, với mỗi tác giả là 1

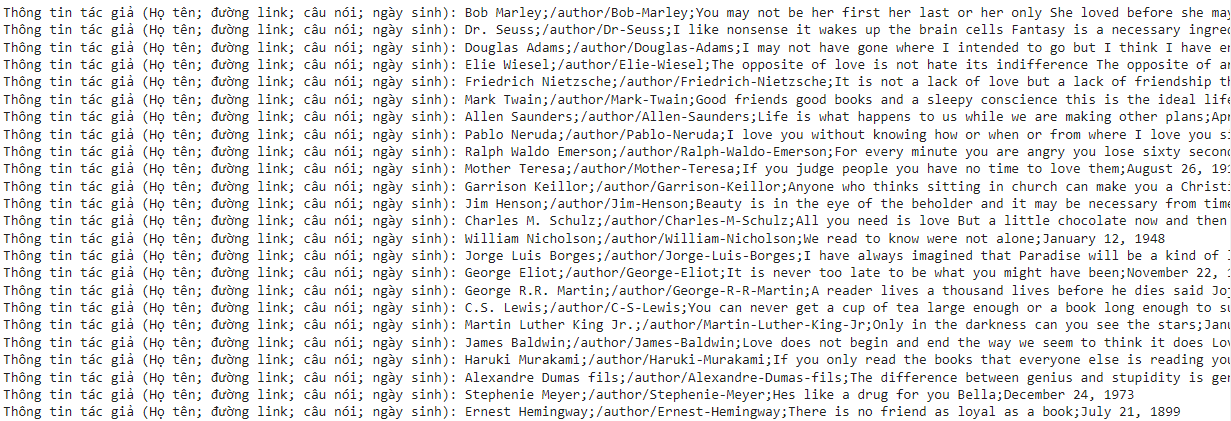
dòng dữ liệu. Bạn được yêu cầu thu thập ít nhất 40 câu nói nổi tiếng từ trang

web trên một cách tự động theo code của các ý trên?

Để crawl đủ dữ liệu 40 dòng của đề bài chúng ta sẽ cần crawl thêm dữ ở những tiếp theo nữa thay vì chỉ crawl trang đầu tiên. Do đó chúng ta sẽ chạy một vòng lặp crawl hết 10 trang của trang web này.



*Hình 1.1.8 lấy thêm 10 trang*



CHƯƠNG II – KHAI PHÁ DỮ LIỆU

2.1 Xử lý dữ liệu - Data Imputation:

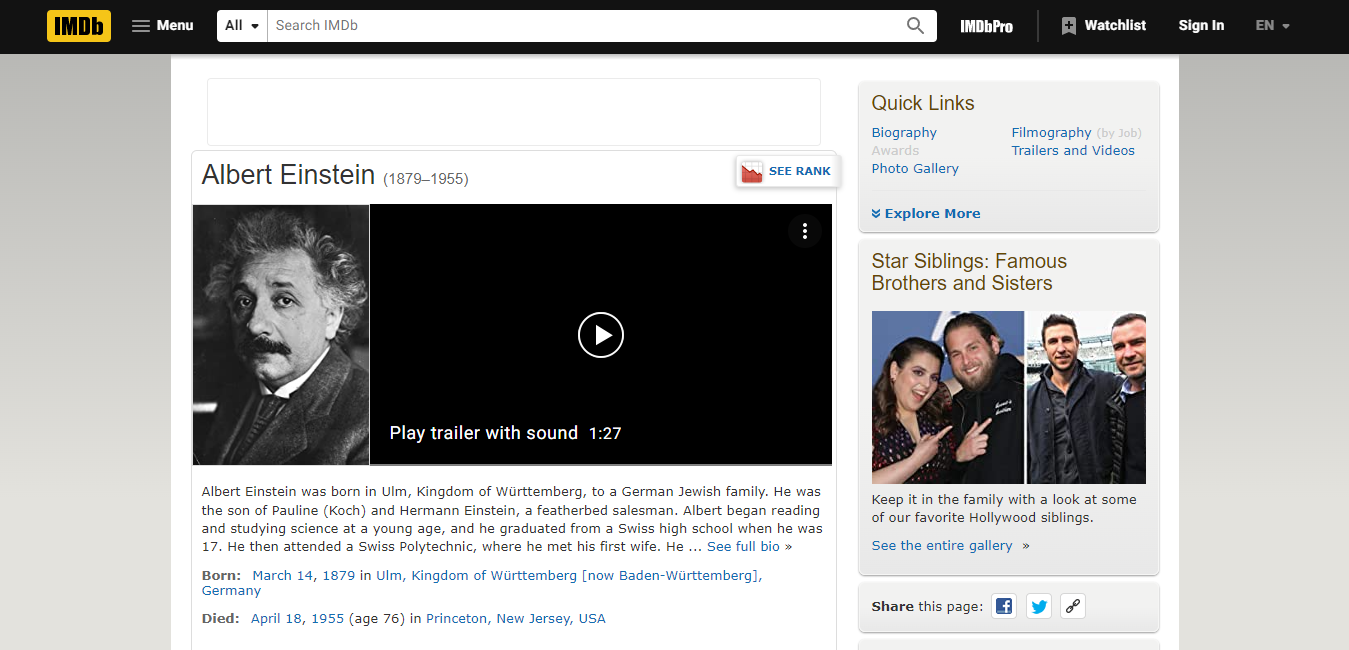
### Đề xuất cách điền cho một số trường ngày sinh chưa có:

Missing data là dữ liệu bị thiếu, được hiển thị như NaN, Nat, Null, N/A, v.v. Missing data xuất hiện do nhiều nguyên nhân như:

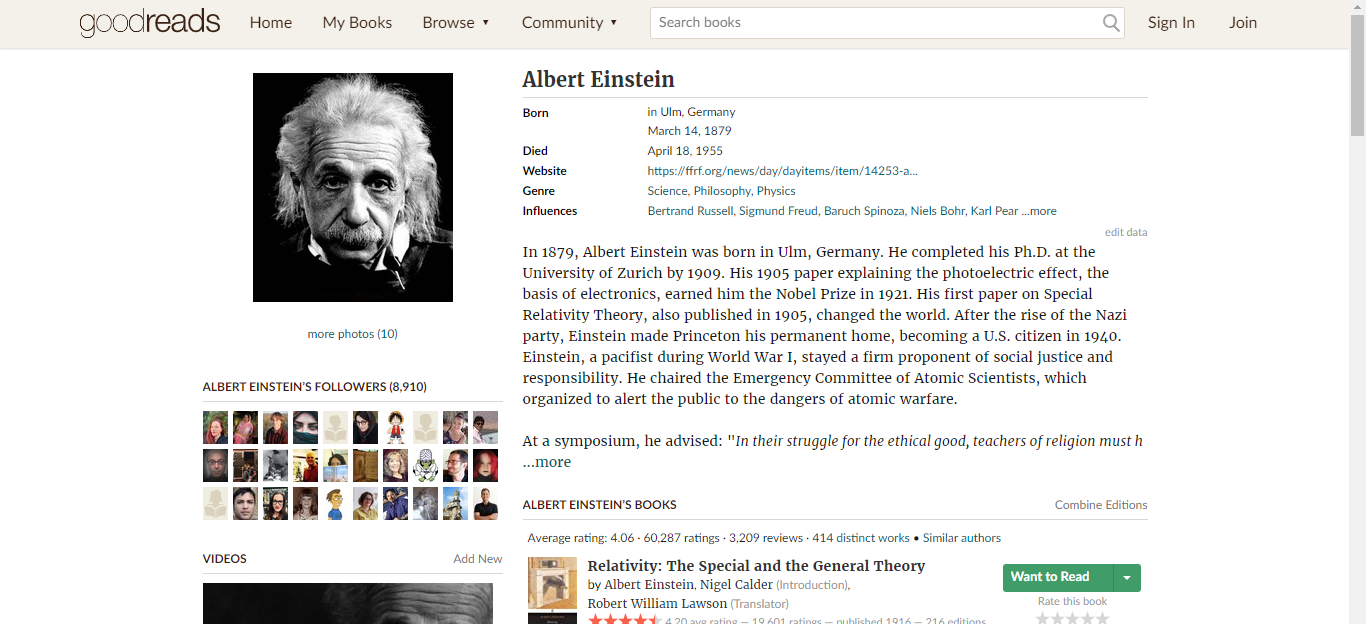
* Người dùng quên điền.
* Dữ liệu bị mất trong quá trình chuyển thủ công từ cơ sở dữ liệu cũ.
* Lỗi của chương trình.
* Thiếu dữ liệu do trích xuất bị sai.
* Trang web bị thiếu trường dữ liệu đó.

Missing data có thể được phân thành 3 loại: Missing at Random (dữ liệu khuyết ngẫu nhiên), Missing Completely at Random (dữ liệu thiếu hoàn toàn ngẫu nhiên) và Missing Not at Random (dữ liệu khuyết không ngẫu nhiên). Trong bài này nếu trường ngày sinh bị thiếu thì nó sẽ là kiểu Missing Completely at Random. Để xử lý trường hợp missing data kiểu này chúng ta sẽ

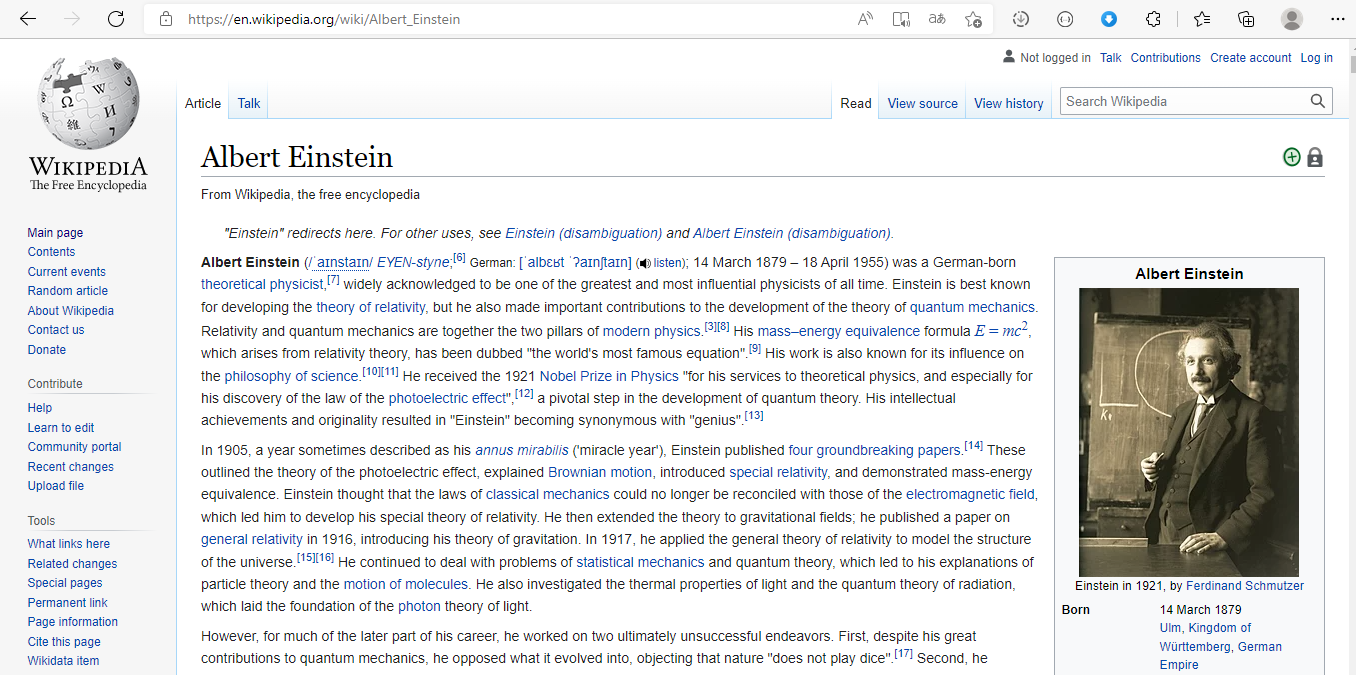
crawl một số nguồn dữ liệu khác cũng có dữ liệu ngày sinh của tác giả như IMDB, Goodreads, Wikipedia.



*Hình 2.1.1 Trang web của tác giả trên IMDB*



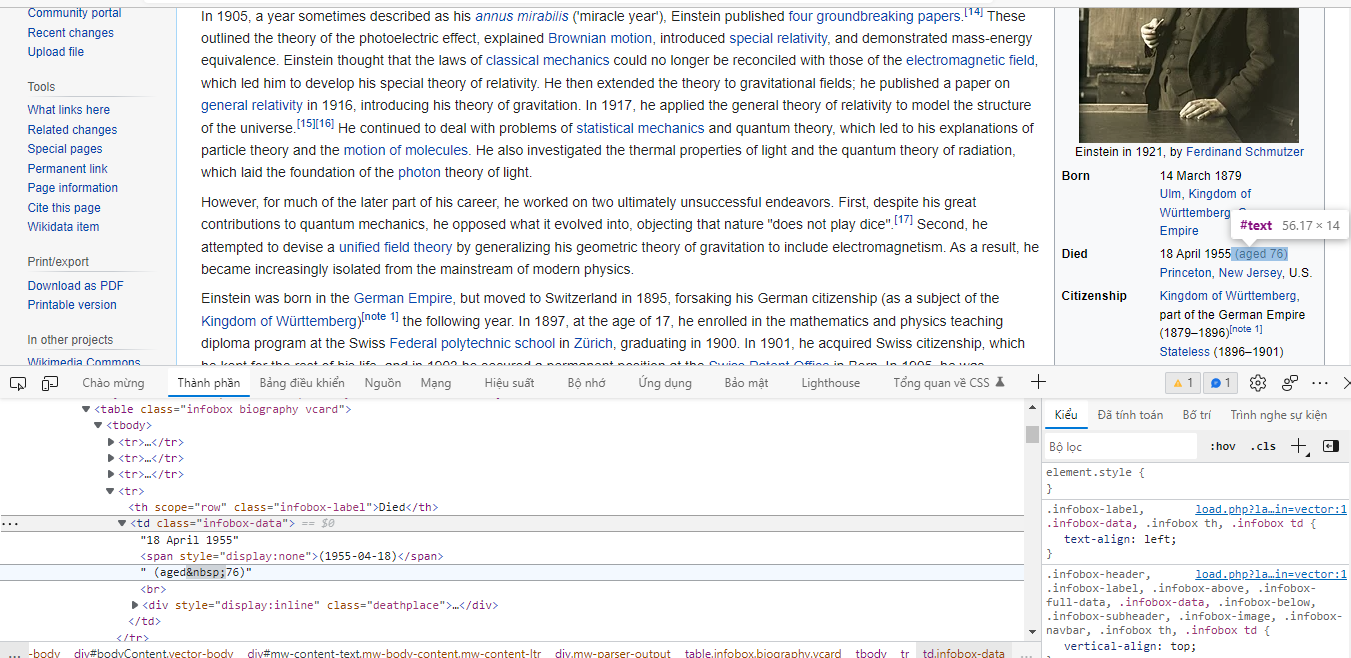
*Hình 2.1.2 Trang web tác giả trên goodreads*



*Hình 2.1.3 Trang web tác giả trên Wikipedia*

### Đề xuất cách điền thêm trường Tuoi(Tuổi):

Để thêm trường tuổi của tác giả chúng ta sẽ dùng nguồn từ wikipedia. Cụ thể là trích ra các thẻ < td class="infobox-data">



*Hình 2.1.4 trích các thẻ có class là infobox-data*

Để tái sử dụng lại code cho nhiều tác giả khác nhau chúng ta sẽ viết hàm chuyên trích dữ liệu tuổi của tác giả trên trang wikipedia.

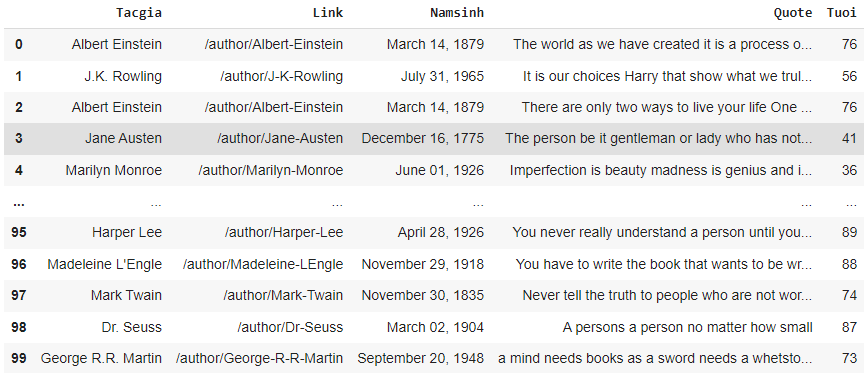


*Hình 2.1.5 Hàm trích dữ liệu tuổi của tác giả trên wiki*

Và để tạo ra đường dẫn wikipedia của từng tác giả chúng ta thêm tên tác giả vào đường link. Ví dụ để đi đến đường dẫn wikipedia của tác **giả Albert Einstein** chúng ta sẽ dùng đường dẫn có dạng như thế này [**https://en.wikipedia.org/wiki/Albert Einstein**](https://en.wikipedia.org/wiki/Albert_Einstein).



*Hình 2.1.6 Thêm đường link tác giả vào*



*Hình 2.1.7 xuất ra kết quả*

Lưu ý: Phải lọc tên tác giả tên có ký tự đặc biệt: André, tác giả phải thêm thông tin phía sau: William Nicholson phải có thêm \_(writer) thì mới ra chính xác tác giả.

2.3 Tiền xử lý dữ liệu - Data Preprocessing:

Tiền xử lí dữ liệu bao gồm nhiều bước trong đó có việc chuyển đổi dữ liệu thô thành các dữ liệu đã được định dạng lại phù hợp cho các mô hình học máy. Tiền xử lí dữ liệu ảnh hưởng cực kì lớn đối với data analyics, big data, data mining, ... bởi vì nếu tiền xử lí không tốt sẽ dẫn đến các lĩnh vực data analyics, big data, data mining dễ bị sai lệch hoặc thiết hiệu quả.

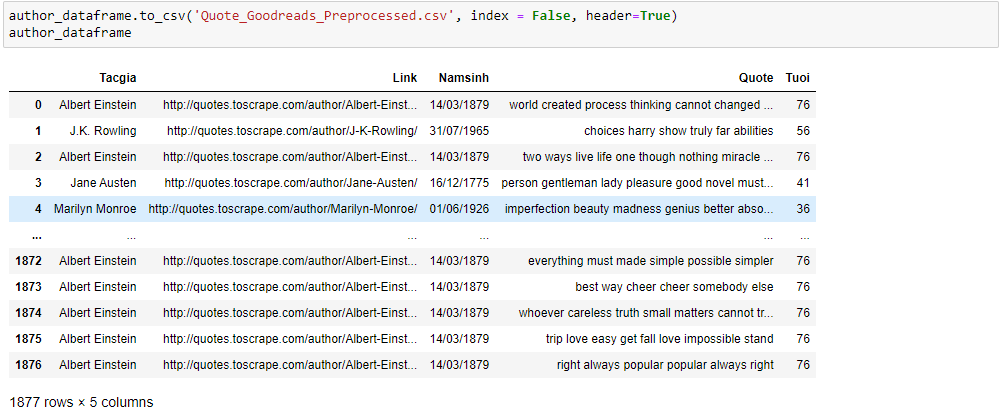
Tiền xử lí trong văn bản sẽ bao gồm việc lọc các ký tự gây nhiễu như ?)!(#&@\*, và các từ có tần suất hiện nhiều trong câu nhưng không đóng góp nhiều ý nghĩa như là các đại từ nhân xưng, on, in, at, but, a, an,... và chuyển chữ viểt hoa thành chữ thường bởi vì học máy sẽ không phân biệt được chữ hoa và chữ thường.

Đối với một ngôn ngữ chuyên dùng để xử lí dữ liệu như python. Python cũng có nhiều module hỗ trợ việc lọc các ký tự gây nhiễu và các từ không đóng góp nhiều ý nghĩa trong câu. Đó chính là moduel nltk và moduel String.



*Hình 2.2.1 viết hàm xử lý tiền văn bản*

Kết quả sau khi dữ liệu đã qua tiền xử lý

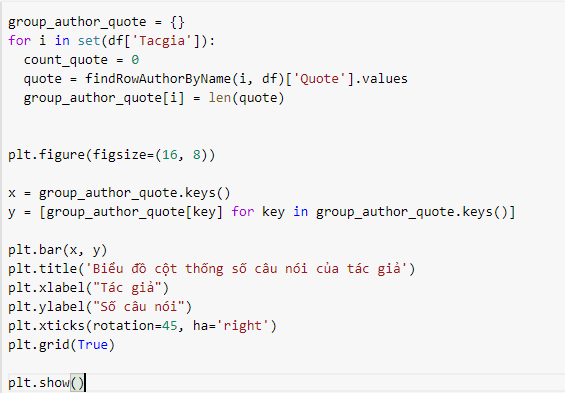


*Hình 2.2.2 kết quả sau khi xử lý*

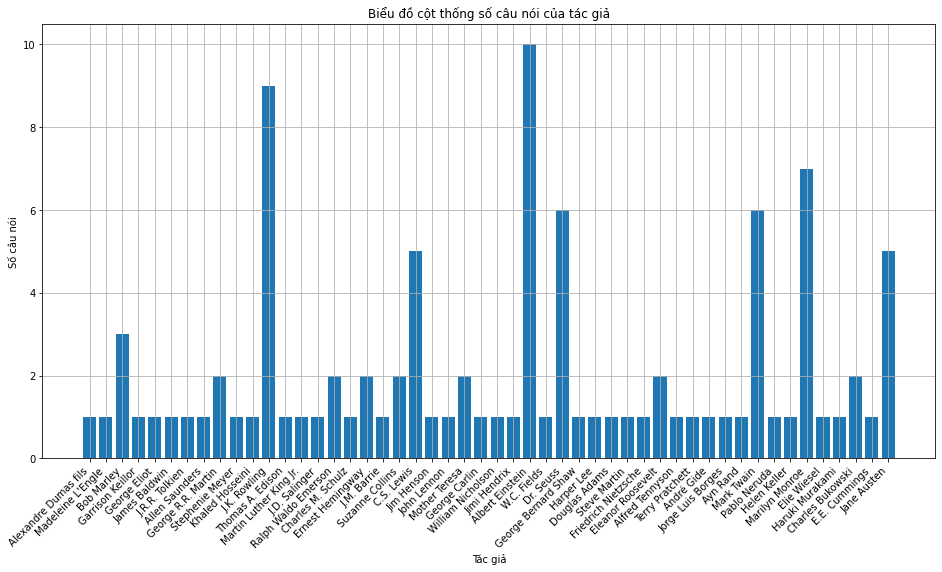
2.4 Khám phá dữ liệu - Data Exploration:

### Thống kê số câu nói của các tác giả:

Để thống kê số câu nói của các tác giả chúng ta sẽ sử dụng biểu đồ cột và sử dụng vòng lặp lấy tên từng tác giả và lưu số câu nói vào một dict và vẽ bằng thư viện matplotlib.

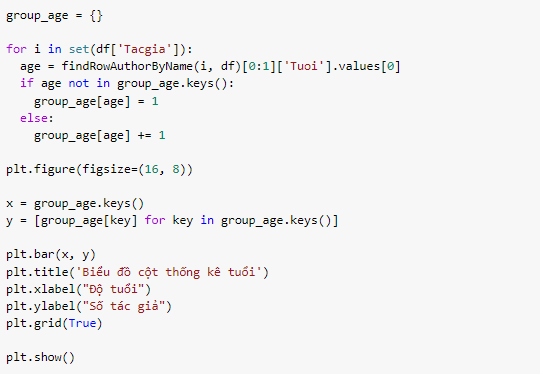


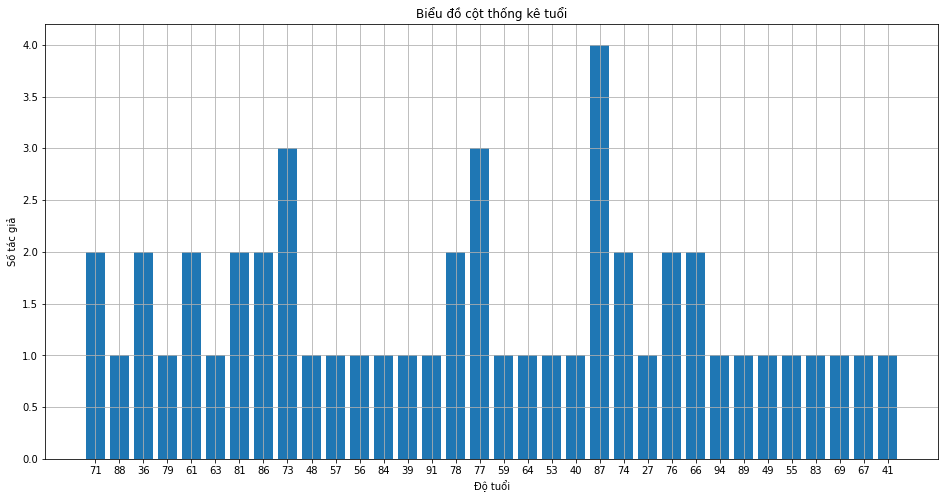
*Hình 2.3.1,Hình 2.3.2 thống kê câu nói của tác giả và biểu đồ thống kê*



### Thống kê tuổi của tác giả:

Để thống kê tuổi của các tác giả chúng ta cũng sử dụng biểu đồ cột và chạy vòng lặp như cách thống kê số câu nói bên trên

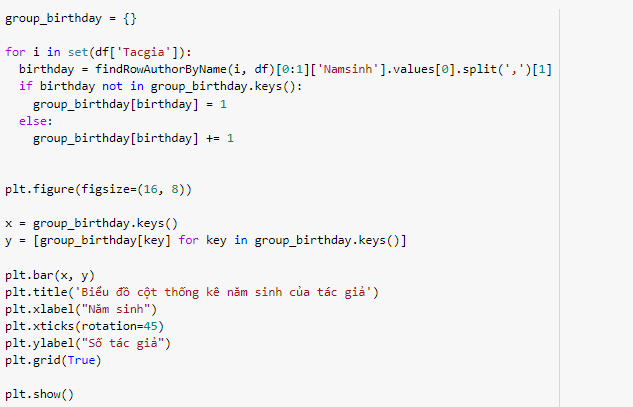


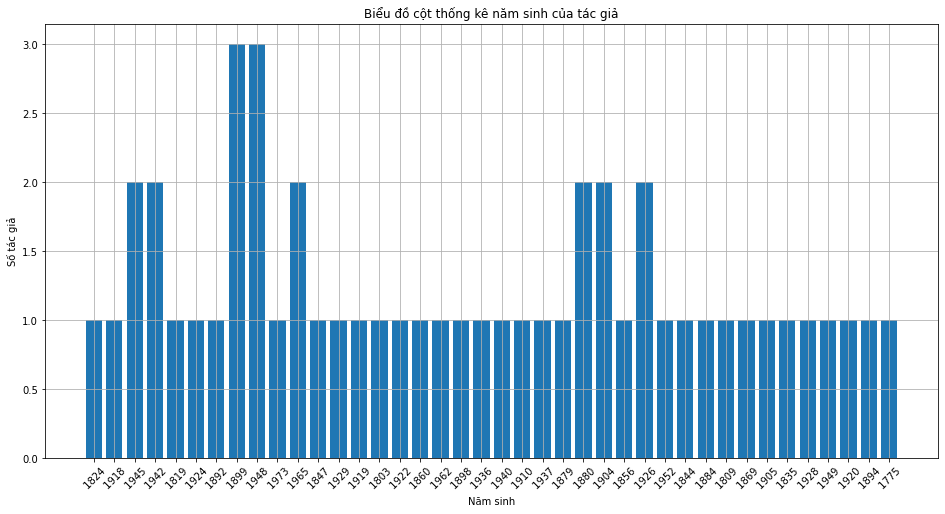


*Hình 2.3.3,Hình 2.3.4 thống kê tuổi và biểu đồ thống kê*

### Thống kê năm sinh của tác giả:

Tương tự như việc thống kê tuổi và thống kê số câu nói của tác giả tuy nhiên thay vì lấy cả ngày sinh chúng ta chỉ cần lấy năm sinh để thống kê:

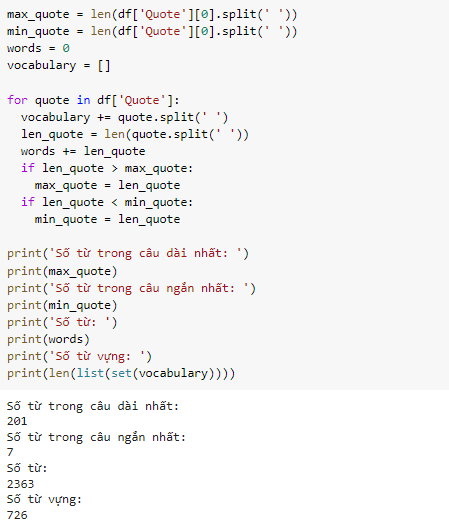




*Hình 2.3.5,Hình 2.3.6 thống kê năm sinh và biểu đồ thống kê*

### Thống kê về câu dài nhất, ngắn nhất, số từ:

Thống kê câu dài nhất và ngắn nhất,số từ chúng ta chỉ việc chạy vòng lặp so sánh với độ dài từng câu và đếm từ sau đó in kết quả ra màn hình.



Hình 2.3.7 Tìm câu dài nhất,ngắn nhất

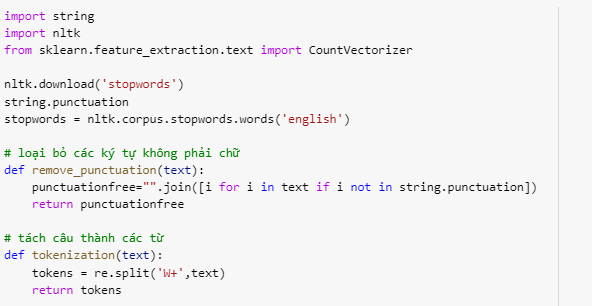
2.5 Trích xuất đặc trưng - Feature Extraction:

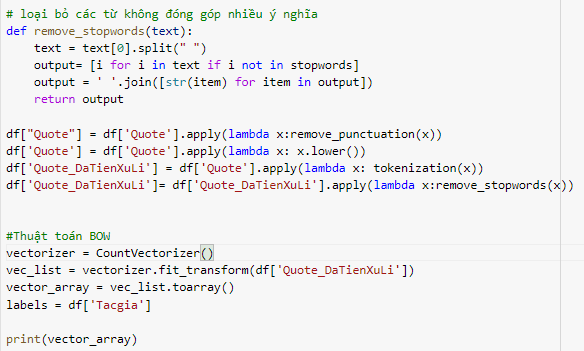
Việc trích xuất đặc trưng rất quan trọng với lĩnh vực học máy do nó ảnh hưởng trực tiếp đến kết quả, hiệu suất học của mô hình. Để xuất đặc trưng văn bản chúng ta cần phải làm 3 bước sau:

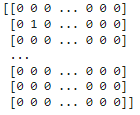
Tiền xử lí trong văn bản sẽ bao gồm việc

* Lọc các ký tự gây nhiễu như ?@\*,)!(#&
* Chuyển từ in hoa sang viết thường (do trong học máy viết hoa và viết thương là 2 từ hoàn toàn khác nghĩa).
* Xóa các dừng từ câu - những từ hay xuất hiện nhiều nhưng không đóng góp nhiều ý nghĩa như là các đại từ nhân xưng.
* Cuối cùng là chuyển đổi dạng văn bản thành dạng vector bằng cách sử dụng kỹ thuật BOW (bag of words).

Để hiện thực các bước trên chúng ta sẽ sử dụng các thư viện String để lọc các ký tự gây nhiễu, thư viện nltk để lọc các dừng từ và sử dụng lớp CountVectorizer() của thư viện sklearn để chuyển từ dạng văn bản sang dạng vector đặc trưng nhiều chiều.







*Hình 2.4.1 các bước trích xuất đặc trưng*

CHƯƠNG III – TỔNG KẾT

3.1 Cài đặt các thư viện yêu cầu:

pip install mwparserfromhell

pip install maplotlib

pip install pandas

pip install nltk

3.2 Tài liệu tham khảo:

1. <https://stackoverflow.com/questions/19324453/add-missing-dates-to-pandas-dataframe>
2. <https://pandas.pydata.org/pandas-docs/version/0.12.0/missing_data.html>
3. <https://ichi.pro/vi/cach-xu-ly-cac-gia-tri-bi-thieu-trong-python-38759789867696>
4. <https://towardsdatascience.com/7-ways-to-handle-missing-values-in-machine-learning-1a6326adf79e>