TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO GIỮA KỲ**

**MÔN KHAI THÁC DỮ LIỆU VÀ KHAI PHÁ TRI THỨC**

**KNOWLEDGE DISCOVERY AND DATA MINING**

**MID-TERM PROJECT**

*Người hướng dẫn*: **PHD. Lê Cung Tưởng**

*Người thực hiện*: **Nguyễn Thị Thảo Như – 51900162**

**Nguyễn Trần Minh Hoa – 51900076**

**Trần Thị Kiều - 51900626**

Khoá    **:    23**

**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 KỲ**

**MÔN KHAI THÁC DỮ LIỆU VÀ KHAI PHÁ TRI THỨC**

**KNOWLEDGE DISCOVERY AND DATA MINING**

**MID-TERM PROJECT**

*Người hướng dẫn*: **PHD. Lê Cung Tưởng**

*Người thực hiện*: **Nguyễn Thị Thảo Như – 51900162**

**Nguyễn Trần Minh Hoa – 51900076**

**Trần Thị Kiều - 51900626**

Khoá    **:    23**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2022**

LỜI CẢM ƠN

Trước tiên, nhóm nghiên cứu chúng em xin gửi lời cảm ơn chân thành đến thầy Lê Cung Tưởng, thầy đã luôn hỗ trợ cho chúng em trong thời gian học tập trên lớp và cuối cùng là báo cáo giữa kỳ này. Thầy luôn giải đáp tất cả các thắc mắc cũng như hướng dẫn tận tình cho chúng em trong thời gian học vừa qua.

Tiếp theo, nhóm chúng em xin gửi lời cảm ơn đến khoa Công nghệ thông tin trường Đại học Tôn Đức Thắng, khoa đã tạo điều kiện cho chúng em có các tiếp xúc thực tế với môn học bằng việc làm các báo cáo nghiên cứu như thế này, giúp chúng em có thêm nhiều kinh nghiệm cho các vị trí việc làm trong tương lai cũng như một lần tổng hợp lại kiến thức đã học được trong thời gian qua.

Cuối cùng, nhóm chúng em xin chúc thầy cô giảng viên và khoa Công nghệ thông tin có thật nhiều sức khỏe, ngày càng phát triển để có thật nhiều thành tích cống hiến cho trường Đại học Tôn Đức Thắng cũng như trong lĩnh vực công nghệ nói chung.

TẬP THỂ NHÓM NGHIÊN CỨU XIN CHÂN THÀNH CẢM ƠN!

BÁO CÁO ĐƯỢC HOÀN THÀNH TẠI

TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG

Chúng em xin cam đoan đây là báo cáo riêng của nhóm chúng em, các nội dung nghiên cứu chưa có bất cứ công bố dưới hình thức nào. Các thông tin từ bài báo cáo đều được nhóm chúng em thu thập từ việc tìm hiểu tham khảo ở các tài liệu có liên quan. Các tài liệu tham khảo đều được nhóm đề cập trong phần “TÀI LIỆU THAM KHẢO”  ở cuối bài báo cáo.

Nếu có bất cứ vi phạm về gian lận hay các vi phạm về bản quyền khoa học, nhóm chúng em xin chịu hoàn toàn trách nhiệm.

Trường Đại học Tôn Đức Thắng hoàn toàn không liên quan đến những hành vi vi phạm tác quyền, bản quyền do chúng em gây ra trong quá trình hoàn thành bài báo cáo (nếu có)

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

*Tác giả*

*(ký tên và ghi rõ họ tên)*

Nguyễn Thị Thảo Như

Nguyễn Trần Minh Hoa

Trần Thị Kiều

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)

**MỤC LỤC**

[LỜI CẢM ƠN 3](#_Toc100351934)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN 5](#_Toc100351937)

[1. The Eclat algorithm: 7](#_Toc100351938)

[2. Implement the crawling task 12](#_Toc100351939)

[2.1. Write a python program to crawl revenue and profit of data in each year and graphing 12](#_Toc100351940)

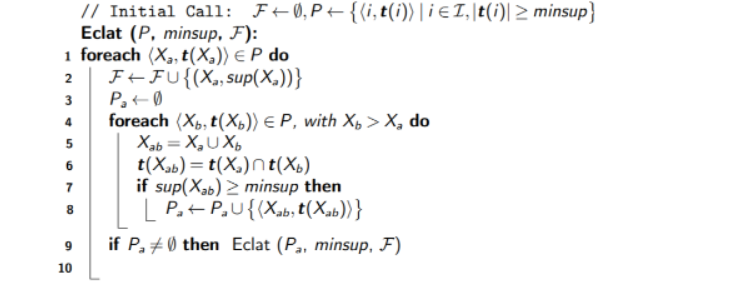
[2.2 Write a python program to crawl a structure of total assets of this company in each quarter and graphing 15](#_Toc100351941)

[3. Crawling and preprocessing text: 18](#_Toc100351942)

[3.1 Wrie a python program to crawl the first 200 comments of this video: 19](#_Toc100351943)

[3.2 Apply text preprocessing techniques and TF-IDF for feature extraction of the first 200 comments: 23](#_Toc100351944)

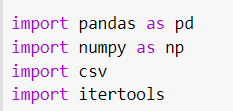
1. The Eclat algorithm:

****

**1.1 Chương trình:**

Thư viện được sử dụng:

* Thư viện pandas
* Thư viện numpy
* Thư viện csv
* Thư viện itertools

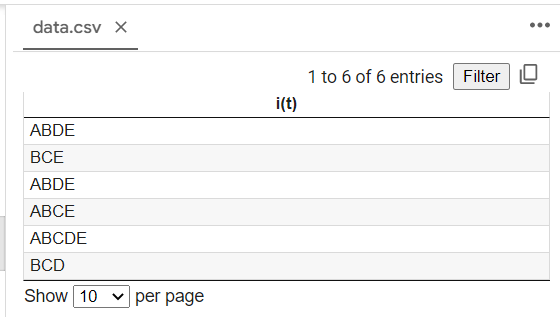


Hình 1.

Các bước tiến hành:

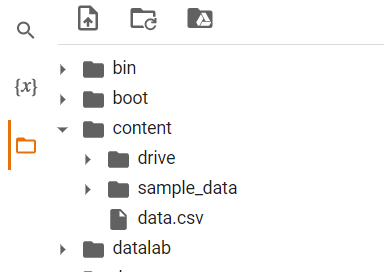
***Bước 1****: Đọc dữ liệu từ file data.csv*

* Ta có file data.csv:



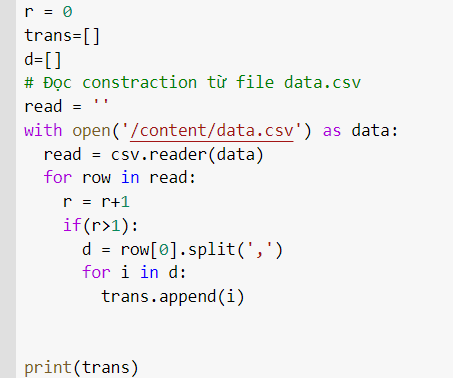
Hình 2.

* File data.csv sẽ được đưa vào drive ở folder content:



Hình 3.

* Source đọc dữ liệu từ file data.csv



Hình 4.

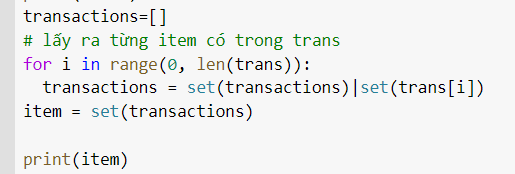
* Kết quả trả về tập hợp trans:



Hình 5.

***Bước 2****: lấy ra số item có trong tập hợp trans:*

* Source code:



Hình 6.

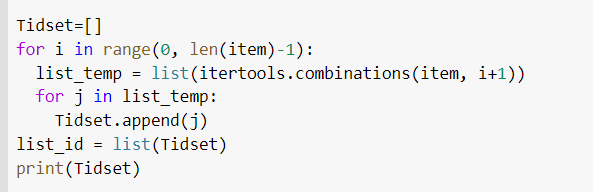
* Kết quả trả về tập cái item của tập trans và được lưu vào tập transactions:



Hình 7

***Bước 3:*** *Kết hợp tất cả các phần tử với nhau:*

* Source code:



Hình 8.

* Kết quả thu được là một tập con và được lưu vào list Tidset:





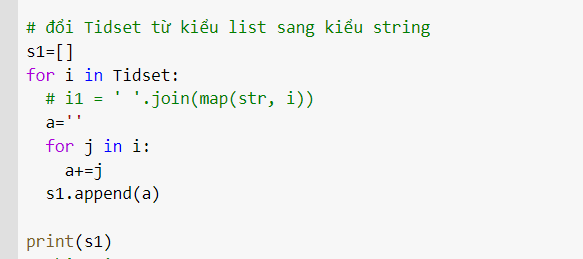




Hình 9.

***Bước 4:*** *Đi tìm id của từng item trong Tidset:*

* Đầu tiên ta phải đổi tập Tidset từ kiểu list sang kiểu string:
* Source code:



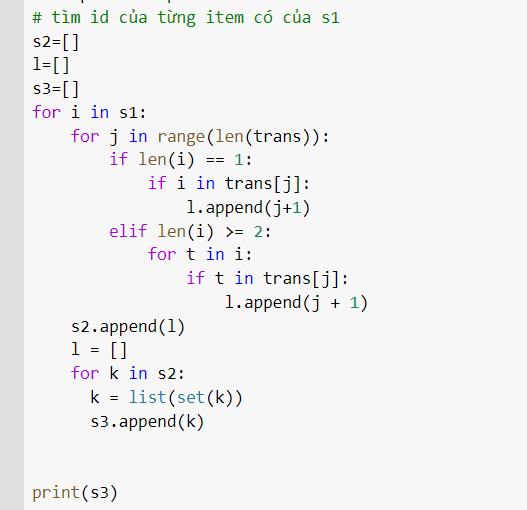
Hình 10.

* Kết quả thu được sẽ lập chuỗi s1 có dạng như sau:



Hình 11.

* Tiếp theo ta đi tìm từng id của từng item trong chuỗi s1:
* Source code:



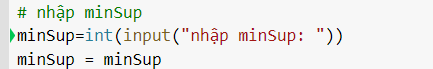
Hình 12.

* Kết quả thu được id của từng item, và sẽ được lưu vào s3:

…..

***Bước 5:*** *xét item thỏa mãn điều kiệm minSup (>=minSup):*

* Nhập minSup từ bàn phím:
* Source code:

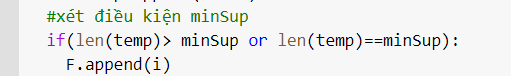


* Kết quả thu được (Ví dụ nhập minSup =2):





* Xét điều kiện thỏa minSup:
* Source code:



* + Kết quả thu được các item thỏa mãn điều kiện minSup (>=minSup) và tập kết quả được lưu vào tập F



2. Implement the crawling task

Thu thập dữ liệu web (hoặc thu thập dữ liệu) được sử dụng để [trích xuất dữ liệu](https://oxylabs.io/blog/what-is-data-extraction) và đề cập đến việc thu thập dữ liệu từ world wide web hoặc trong các trường hợp thu thập dữ liệu - bất kỳ tài liệu, tệp nào, v.v. Chương trình mà thực hiện công việc này gọi là **web crawler**.

# **2.1. Write a python program to crawl revenue and profit of data in each year and graphing**

Trong trường hợp này chúng ta có thể sử dụng nhiều thư viện khác nhau để Crawl dữ liệu về. Thư viện [BeautifulSoup](https://www.crummy.com/software/BeautifulSoup/bs4/doc/) là một thư viện của Python cho phép chúng ta lấy dữ liệu từ HTML đơn giản và hiệu quả.

**2.1.1 Tìm hiểu về crawling dữ liệu từ một trang web**

Truy cập đến các thành phần HTML có chứa nội dung mà chúng ta cần thu thập dữ liệu.

Trang web sử dụng để crawling dữ liệu chứng khoáng của nhóm chúng em là: <https://www.cophieu68.vn/index.php>

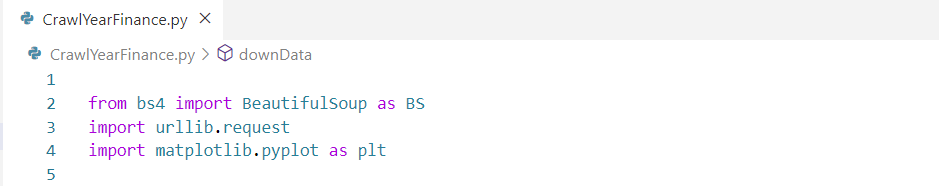
**2.1.2 Viết chương trình Python**

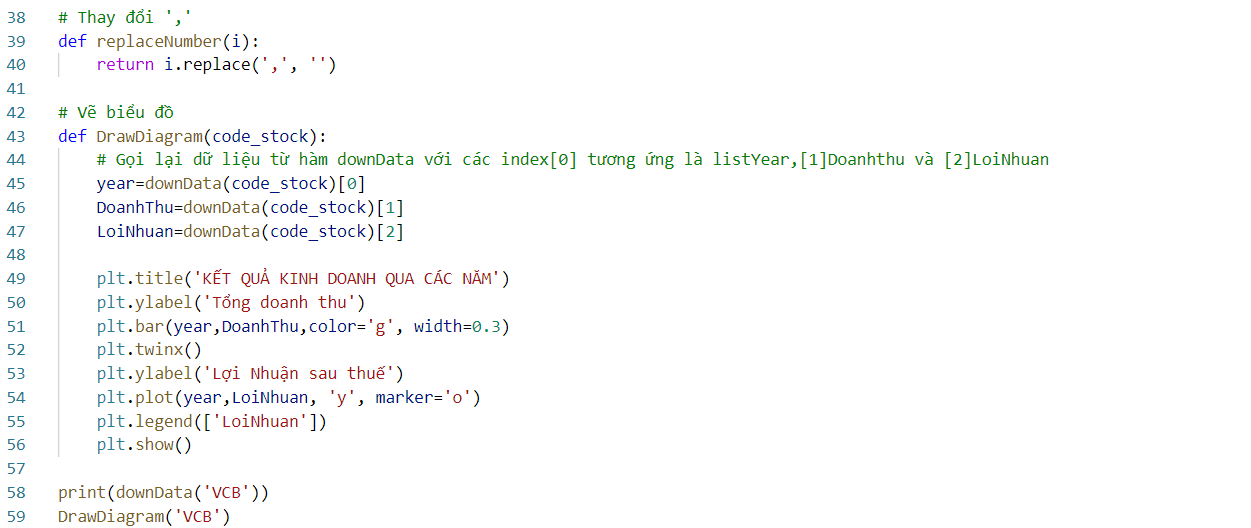
Đầu vào của chương trình là mã chứng khoáng nên chúng ta sẽ truyền vào biến code\_stock.

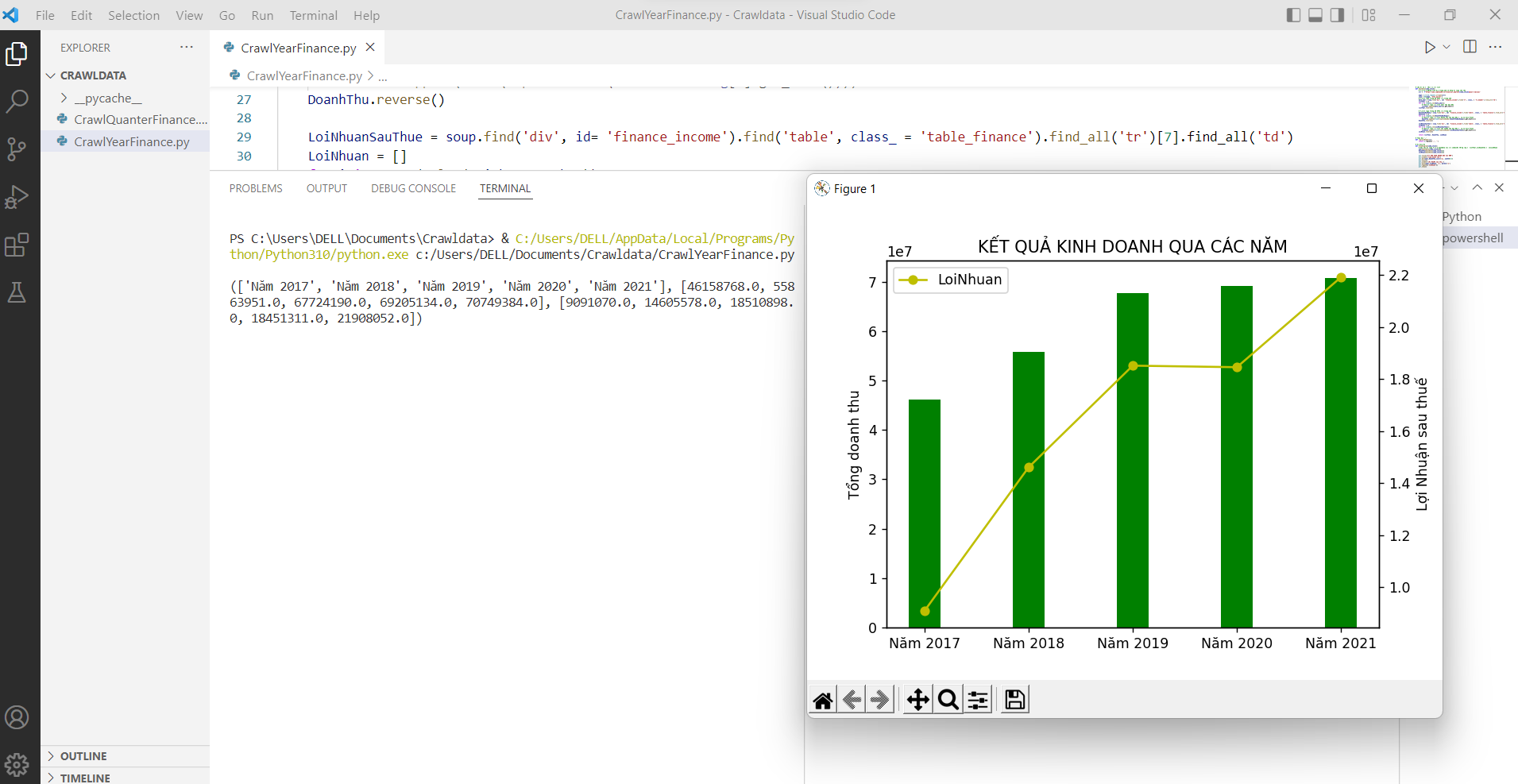
Đầu tiên để thực hiện lấy dữ liệu về ta phải truy vấn đề phần tử HTML có chưa dữ liệu cần lấy. Sau khi lấy được dữ liệu thì nó sẽ được append() vào biến rỗng đã được tạo trước đó để sử dụng số liệu đó cho phần phân tích dữ liệu thành biểu đồ.

Hàm downData(code\_stock ) được sử dụng để lấy dữ liệu về và lưu vào các list rỗng đã tạo ra.

Hàm DrawDiagram(code\_stock) được sử dụng để vẽ biểu đồ bằng cách gọi lại dữ liệu của các list ở hàm downData(code\_stock )



   
 **2.1.3 Kết quả thực thi**



# **2.2 Write a python program to crawl a structure of total assets of this company in each quarter and graphing**

**2.2.1 Viết chương trình Python**

Tương tự với bài 2.1, việc crawling dữ liệu từ trang web ở bài 2.2 vẫn phải truy vấn đến phần tử HTML tương ứng để lấy ra các thành phần dữ liệu chúng ta cần. Sau đó append() chúng vào các list rỗng và sử dụng nó cho quá trình vẽ biểu đồ.

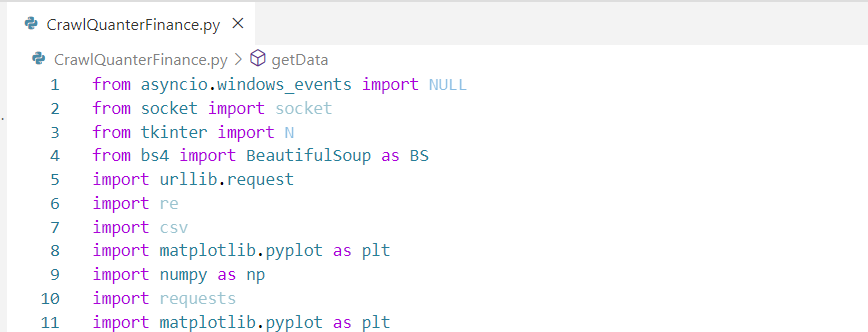
Đối với bài 2.2 nhóm chúng em xây dựng 3 hàm đó là hàm downData(code\_stock, year), getData(code\_stock) và hàm drawChart(code\_stock).

Hàm downData được sử dụng để lấy dữ liệu từ trang web về và lưu vào các biến Temp. Vì dữ liệu ở đây có 2 dạng đó là đối với các năm có dữ liệu thì sẽ có 3 bảng và các năm không có dữ liệu sẽ có một bảng cuối cùng đó là Tăng trưởng tài chính. Vì bảng có dữ liệu thực hiện chạy từ i =2 (bỏ cột với tiêu đề “Xem chi tiết Cân đối kế toán” và cột “Năm 2021”) nên nếu dữ liệu đúng thì sẽ bắt đầu lấy từ Quý 1.... Nếu dữ liệu bắt đầu bằng “ Năm....” hoặc độ dài của Quý bằng 0 thì kết quả sẽ trả về 0. Sau đó dữ sẽ thực hiện append() dữ liệu vào biến QuanterTemp được tạo để lấy dữ liệu Quý.

Thực hiện tương tự với các thành phần cần lấy ta sẽ được các biến Temp có dữ liệu ở hàm downData.

Hàm getData kiểm tra điều kiện xem năm lấy vào có lớn hơn 2000 hay không vì chỉ có những năm lớn hơn năm 2000 mới có dữ liệu nếu đúng dữ liệu sẽ được append() vào các list rỗng được tạo ở hàm getData.

Hàm drawChart được xây dựng để vẽ biểu đồ với các số liệu được lấy từ hàm getData.



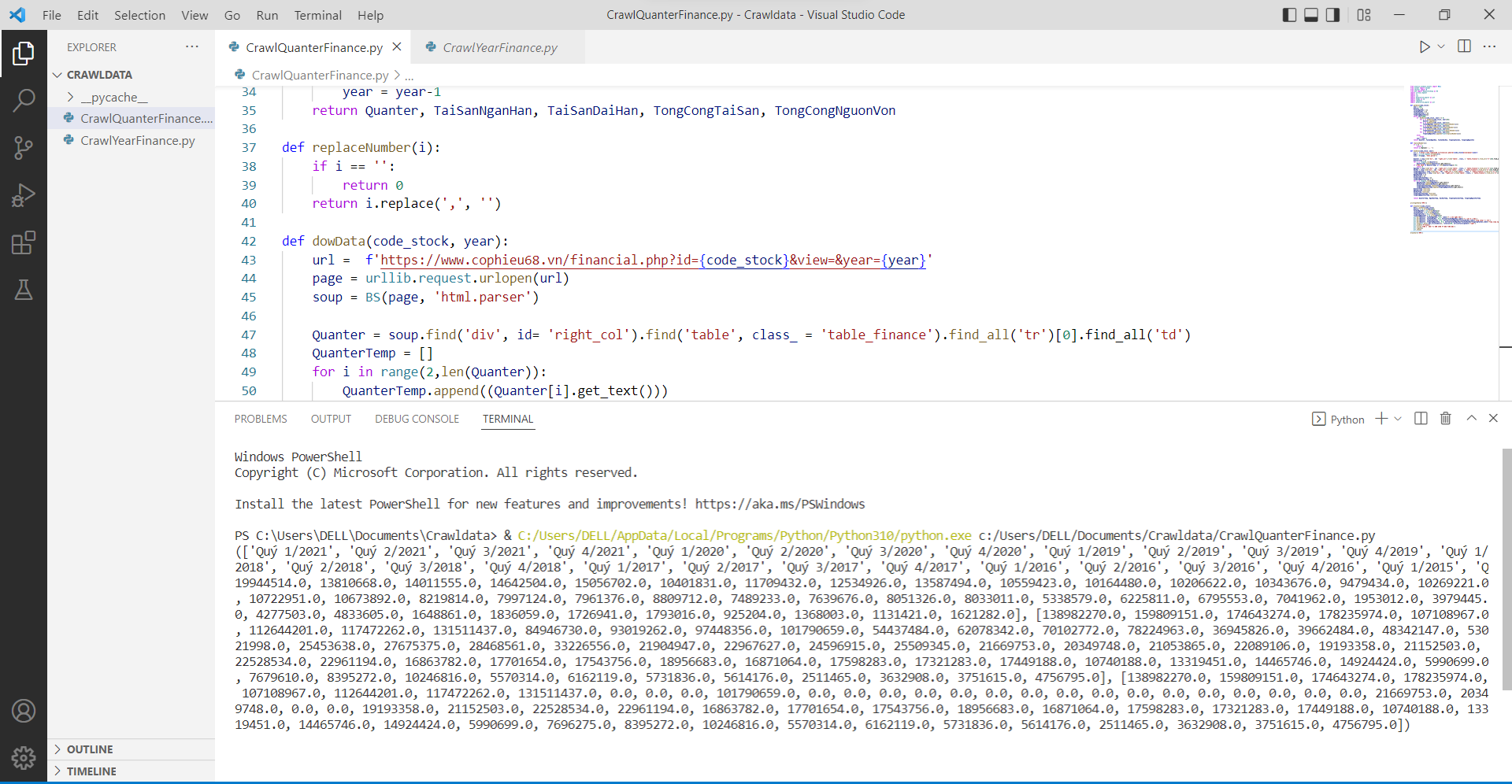


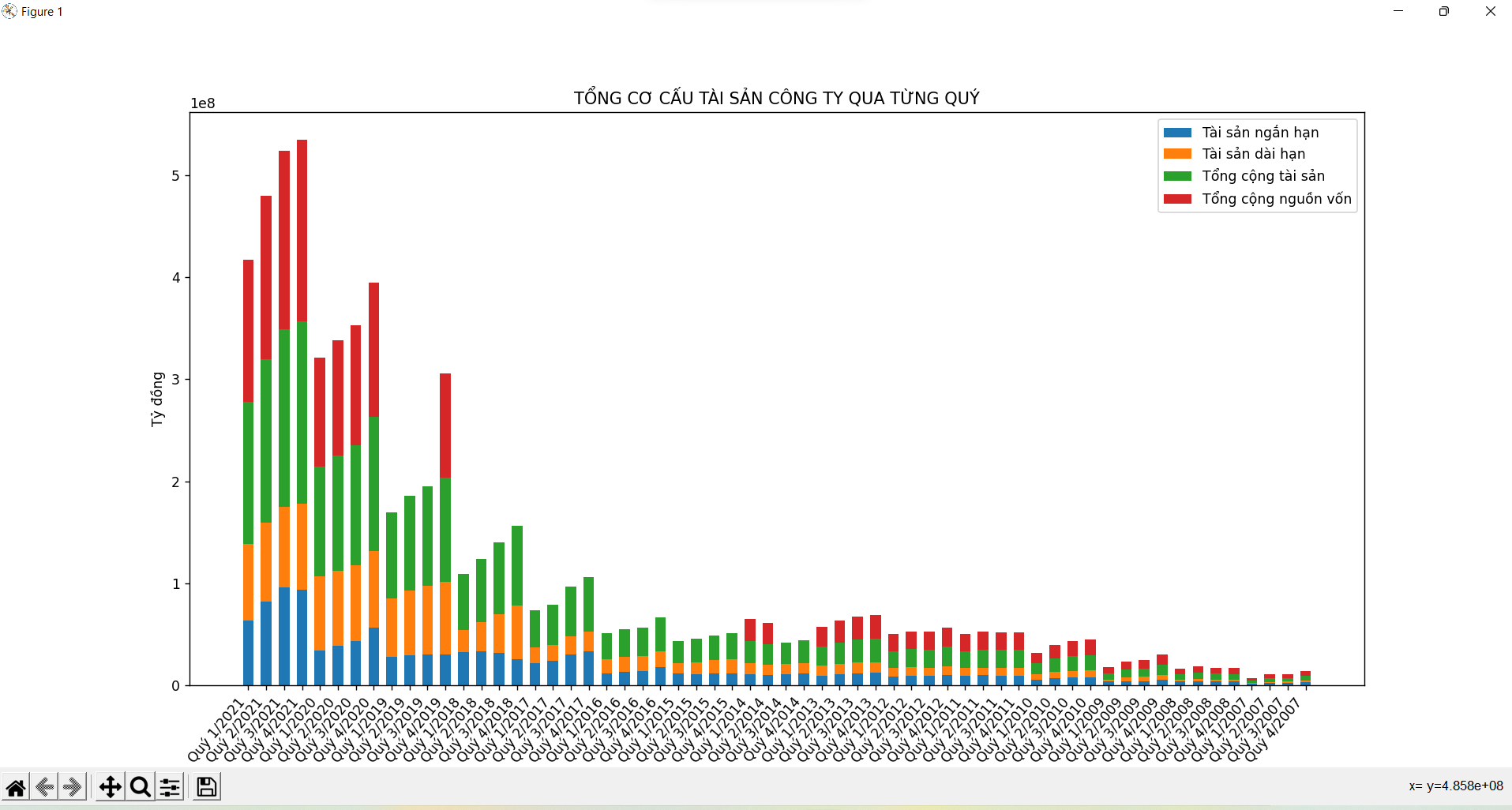






**2.2.2 Kết quả trả về**





3. Crawling and preprocessing text:

Crawling (cào dữ liệu) là một quá trình thu thập dữ liệu từ những trang web và xử lý nó thành những định dạng phù hợp.

Preprocessing text (tiền xử lý văn bản) là một phương thức để làm sạch dữ liệu văn bản và làm cho nó sẵn sàng để cung cấp dữ liệu để xử lý. Dữ liệu văn bản thường chứa nhiều giá trị nhiễu như biểu tượng cảm xúc, dấu chấm câu, ký tự đặc biệt, …. Nên nhiệm vụ của tiền xử lý văn bản chính là loại bỏ những giá trị nhiễu. Một số bước cần có của tiền xử lý là:

* Loại bỏ các dấu đặc biệt như .,!$()\*%@#
* Loại bỏ URLs
* Lower casing
* Tokenization
* Stemming

**TF-IDF** (Term Frequency – Inverse Document Frequency) là 1 kĩ thuật sử dụng trong khai phá dữ liệu văn bản. Trọng số này được sử dụng để đánh giá tầm quan trọng của một từ trong một văn bản. Giá trị cao thể hiện độ quan trọng cao và nó phụ thuộc vào số lần từ xuất hiện trong văn bản.

# **3.1 Wrie a python program to crawl the first 200 comments of this video:**

Chúng ta có nhiều phương pháp để có thể crawl dữ liệu từ trang web về, nhưng nổi bật là hai phương pháp sử dụng Selenium và API.

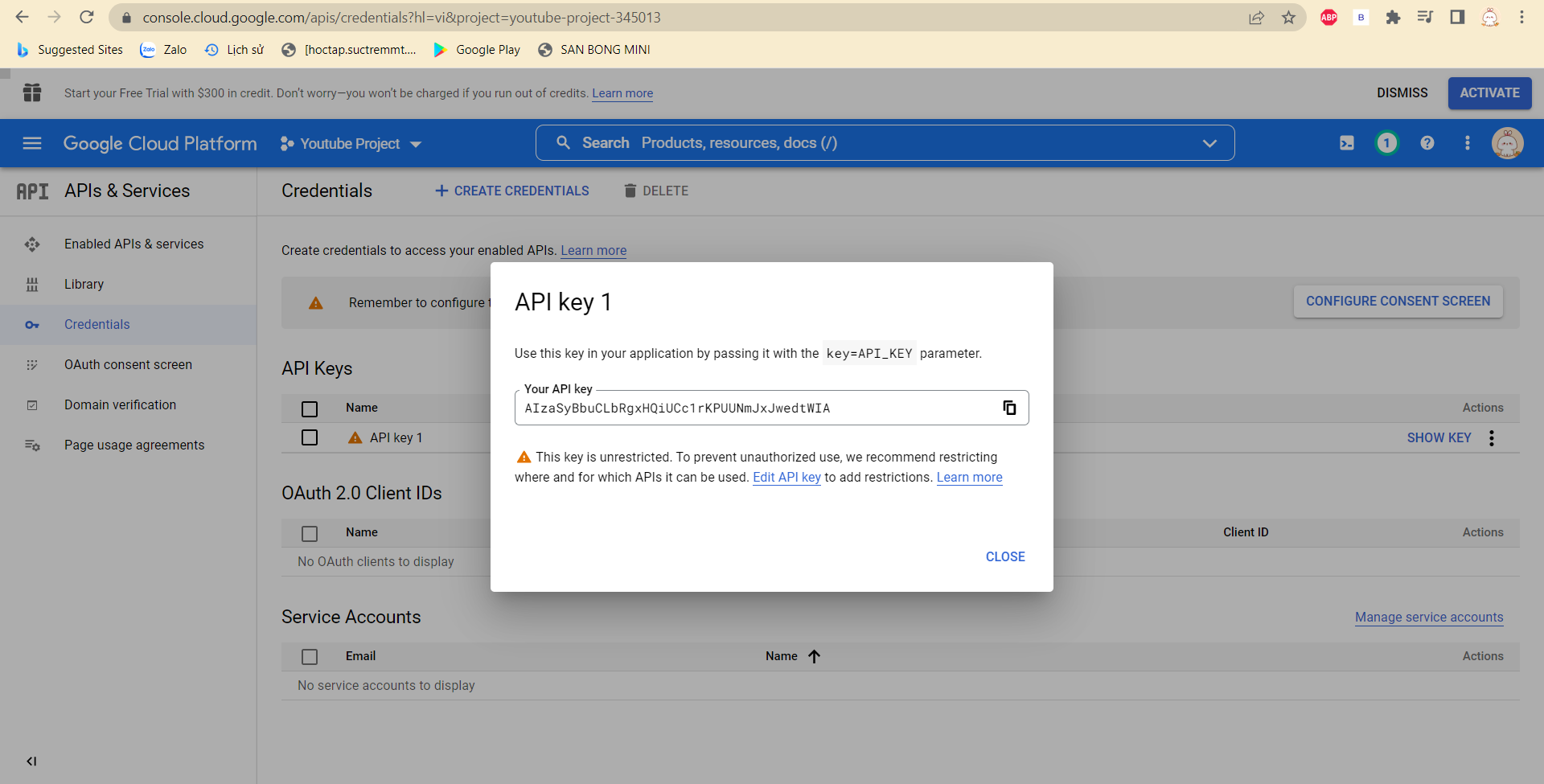
* Đối với Selenium, đây là một công cụ tự động hóa trình duyệt web, tương tác với các nền tảng không cung cấp API, cũng như để thu thập thông tin web (web crawling). Selenium hỗ trợ nhiều ngôn ngữ như: Python, Java, C#, Ruby,…Dễ dàng tích hợp với Maven, Jenkins, Docker
* Đối với API, đây là một hình thức đơn giản để chúng ta có thể tiếp cận đến chương trình nhằm liên lạc hoặc trao đổi dữ liệu với nhau. API cho phép các công ty mở dữ liệu và chức năng của ứng dụng của họ cho bên thứ ba để tận dụng dữ liệu và chức năng của nhau.

Đối với bài tập này, nhóm em thực hiện phương pháp sử dụng API của Youtube để lấy dữ liệu và sau đó tải dữ liệu này vào Dataframe Pandas, tiếp theo là tiến hành phân tích dữ liệu (text preprocessing)

**3.1.1 Tạo khóa API:**

Tạo khóa API cũng như một hình thức đăng nhập để có thể truy cấp dữ liệu Youtube.

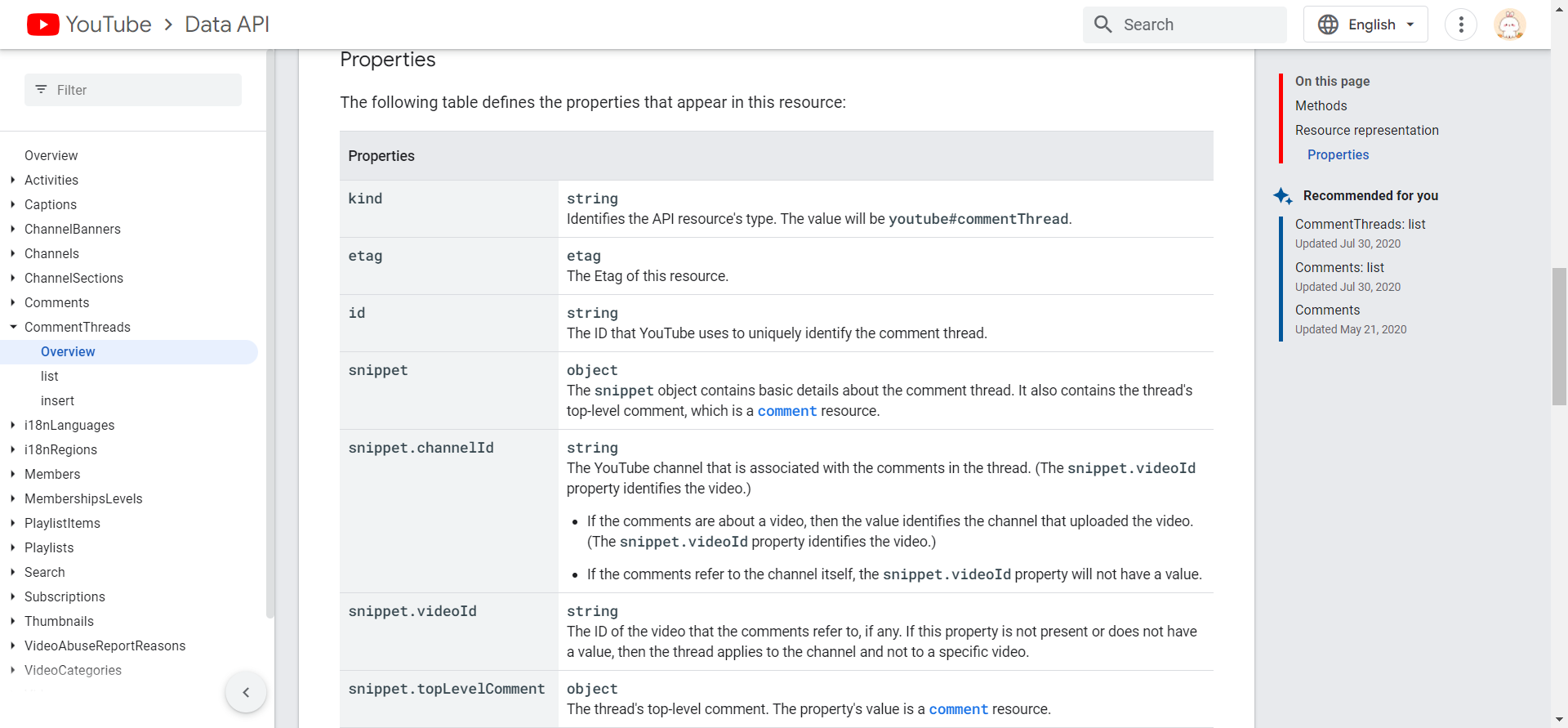
Để thực hiện tạo khóa API, chúng ta truy cập vào trang web <https://console.cloud.google.com/apis/dashboard?hl=vi> sau đó thực hiện kích hoạt Youtube Data API v3 và tiến hành tạo khóa.

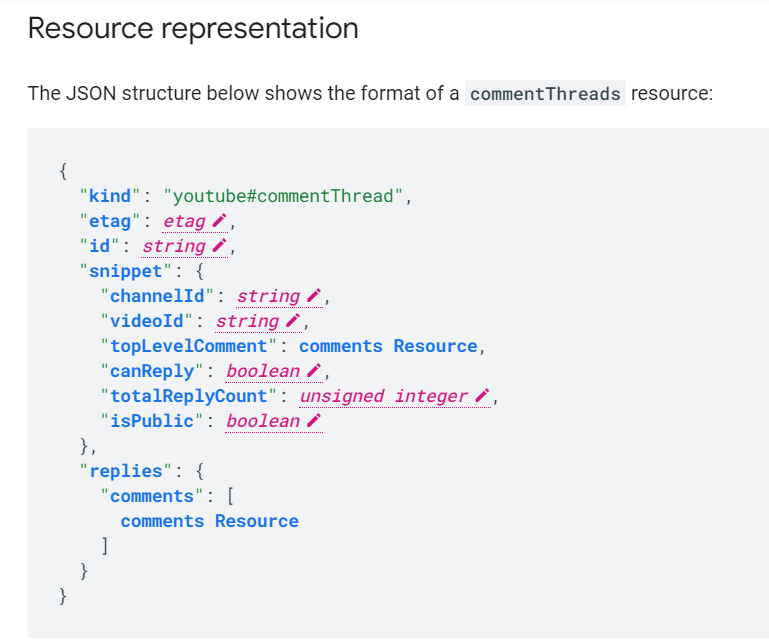


**3.1.2 Tìm hiểu về API Youtube:**

Truy cập vào đường dẫn sau: <https://developers.google.com/youtube/v3/docs/commentThreads>

Chúng ta sẽ xem xét các phần trong mục Comment Thread để tiến hành tìm ra các dữ liệu mà chúng ta cần lấy về để thực hiện xử lý.





**3.1.3 Viết chương trình:**

Đầu tiên, để thực hiện chương trình thì ta phải lấy được hai dữ liệu cần thiết là: API keys và ID của video Youtube mà ta cần crawling dữ liệu.

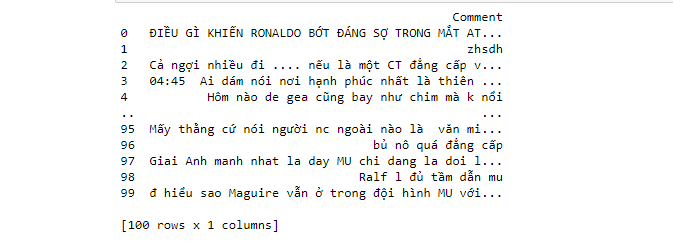
Thay thế API keys vào biến “ *api\_key* ” và id của video vào biến “ *ID* ”

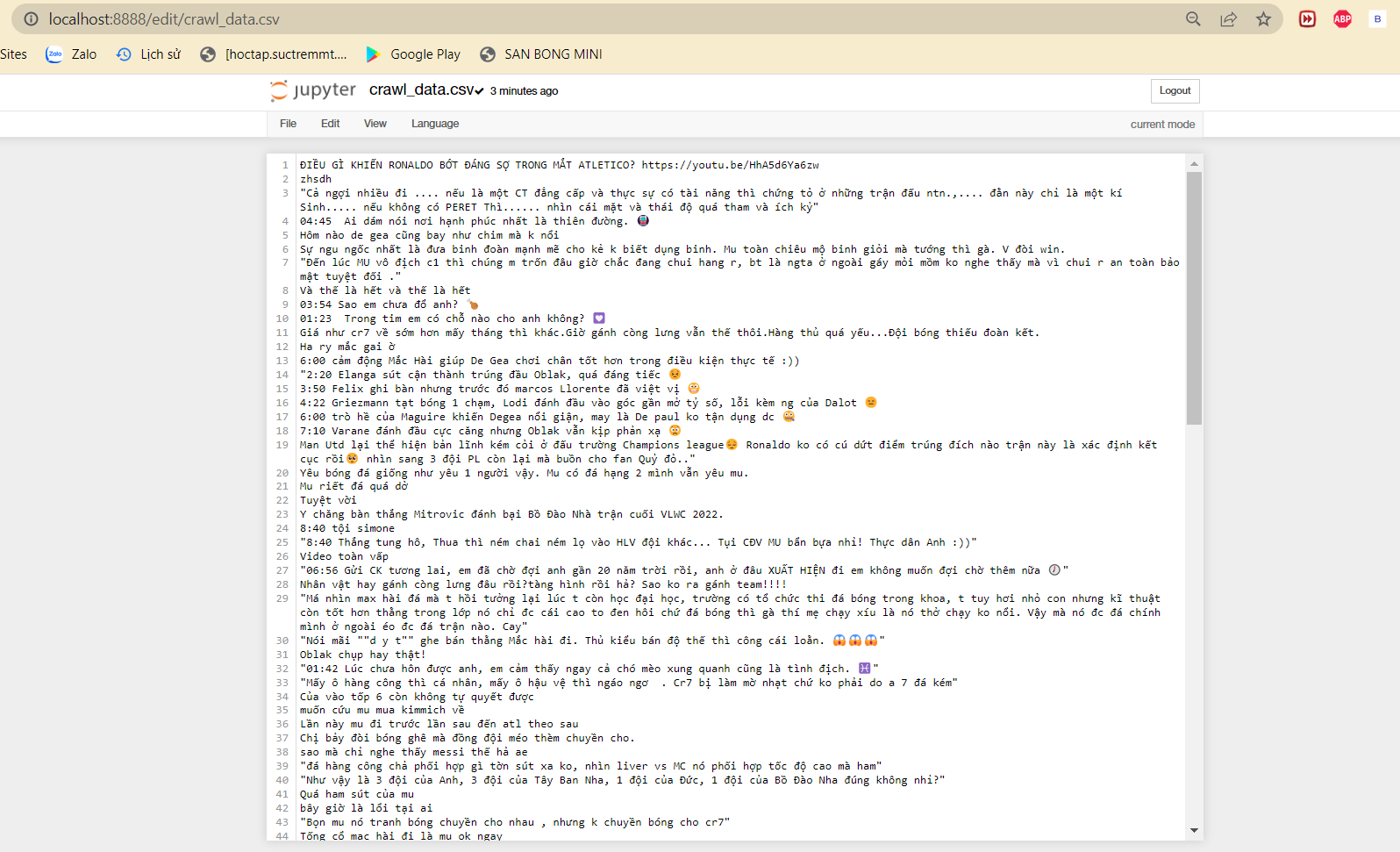
Mô tả tổng quan về hàm *scape\_comments()* :

* Input là: *api\_key, id*  ; Output: *file csv, df[0:200]*
* Dựa vào hình trên, chúng ta có thể thấy để lấy được dữ liệu comment thì ta phải sử dụng *commentThreads* và cụ thể *part = snippet* bởi vì ở đây có chứa một số dữ liệu cần thiết đến như: *comment* nằm trong *topLevelComment*, *publishedAt*, *authorDisplayName* ,….
* Khởi tạo một list rỗng *box* để chứa dữ liệu.
* Đưa dữ liệu *Comment* vào *dataframe*
* Từ 200 dòng dữ liệu đầu tiên của *dataframe* chúng ta xuất ra file *csv*
* Hàm *scape\_comments()* sẽ trả về một dataframe chứa 200 dòng comment để dễ dàng kiểm tra và sử dụng cho bài kế tiếp.



**3.1.4 Kết quả trả về:**





# **3.2 Apply text preprocessing techniques and TF-IDF for feature extraction of the first 200 comments:**

**3.2.1 Text preprocessing:**

**3.2.1.1 Chương trình:**

Để làm sạch dữ liệu, sử dụng 5 hàm với những chức năng khác nhau: xóa icon, xóa ký tự đặc biệt, xóa số, viết hoa thành viết thường, phân tách từ ngữ có nghĩa.

Hàm *remove\_emoji()* có chức năng xóa (thay thế) các ký tự trong chuỗi truyền vào và lọc bỏ đi những ký tự đặc biệt.

Input: string

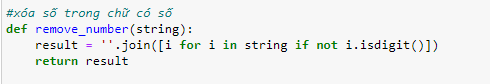
Output: string



Hàm *remove\_numer()* kiểm tra từng thành phần trong chuỗi để loại bỏ số ra khỏi chuỗi.

Input: string

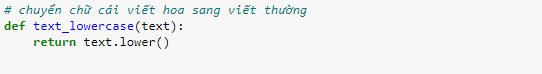
Output: string



Hàm *text\_lowercase()* chuyển ký tự viết hoa sang viết thường.

Input: string

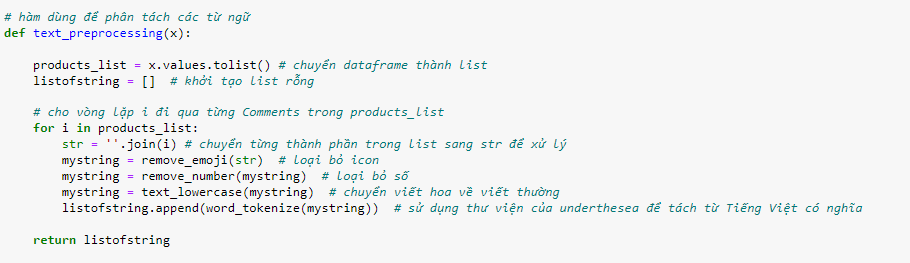
Output: string



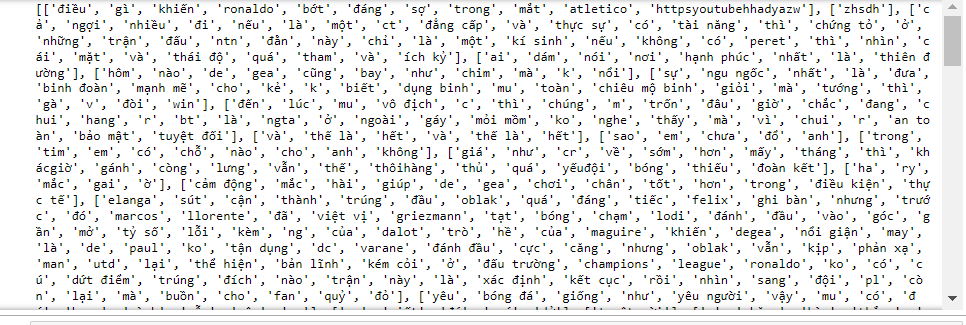
Hàm *text\_preprocessing()* có chức năng là gọi lại các hàm trên để làm sạch dữ liệu và cuối cùng là phân tách các từ Tiếng Việt có nghĩa bằng cách sử dụng thư viện underthesea.

Mô tả cụ thể:

* Input: dataframe
* Chuyển đổi dataframe thành list, sau đó cho vòng lặp chạy trong từng thành phần của list để dùng các hàm đã nêu trên loại bỏ các dữ liệu nhiễu và thêm vào một list mới.
* Output: list



**3.2.1.2 Kết quả:**

****

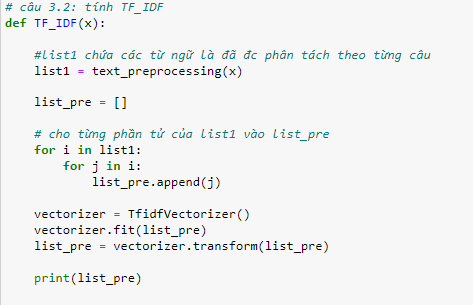
**3.2.2 TF-IDF for feature extraction:**

**3.2.2.1 Chương trình:**

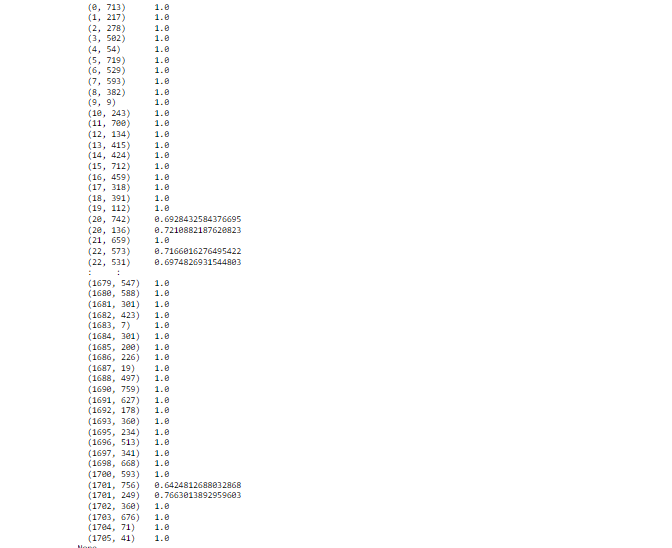
Hàm *TF\_IDF()* sử dụng lại kết quả của hàm *text\_preprocessing()* để tiến hành tính giá trị tf-idf của từng từ.

Mô tả hàm chi tiết:

* Gọi hàm *text\_preprocessing()* trả kết quả vào list1
* Tạo một list rỗng để có thể nối các từ của từng list nhỏ trong list1 lại với nhau tạo thành một list lớn.
* TfidfVectorizer() : hàm chuyển đổi văn bản đầu vào thành các vector dùng để tính toán.
* fit() : lấy các từ trong list và IDF
* transform(): thực hiện chuyển đổi list sang ma trận

****

**3.2.2.2 Kết quả:**

****

***TÀI LIỆU THAM KHẢO***

<https://www.analyticsvidhya.com/blog/2021/06/text-preprocessing-in-nlp-with-python-codes/>

<https://scikitlearn.org/stable/modules/generated/sklearn.feature_extraction.text.TfidfVectorizer.html>

[https://viblo.asia/p/web-crawling-voi-thu-vien-beautifulsoup1VgZvNGOZAw?fbclid=IwAR35aa45DbzaOP5br\_KZkuDlVUeoYMGcwX4PQCo1iaKjIEQLxnNFIhbHKA](https://viblo.asia/p/web-crawling-voi-thu-vien-beautifulsoup-1VgZvNGOZAw?fbclid=IwAR35aa45DbzaOP-5br_KZkuDlVUeoYMGcwX4PQCo1iaKjIEQLxnNFIhbHKA)

<https://atpacademy.vn/crawl-data-la-gi-huong-dan-xuat-du-lieu-tu-1/>