|  |
| --- |
| **BỘ GIÁO DỤC VÀ ĐÀO TẠO**  **TRƯỜNG ĐẠI HỌC SƯ PHẠM KỸ THUẬT TP.HỒ CHÍ MINH** |
| **KHOA CÔNG NGHỆ THÔNG TIN**  ****  **Logo ĐH Sư Phạm Kỹ Thuật TP Hồ Chí Minh-HCMUTE**  **BÁO CÁO ĐỒ ÁN CUỐI KỲ**  **ĐỀ TÀI: PHÂN TÍCH DỮ LIỆU CHỨNG KHOÁN**  **CÓ SỬ DỤNG LTSM DỰ ĐOÁN**  **Môn học: Phân Tích Dữ Liệu**  **GVHD: ThS. Nguyễn Văn Thành**  Nhóm sinh viên thực hiện:Nhóm 1   |  |  | | --- | --- | | Đinh Đức Nguyên Vũ | 20128171 | | Nguyễn Tấn Sương | 21133078 | | Bùi Thị Huỳnh Hân | 21133039 | | Phan Khải Huyền | 21133041 | |

**DANH SÁCH THÀNH VIÊN THAM GIA**

**THỰC HIỆN ĐỀ TÀI VÀ VIẾT BÁO CÁO**

**Môn: Phân Tích Dữ Liệu -** *HỌC KÌ II – NĂM HỌC 2024 – 2025*

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **HỌ VÀ TÊN** | **MSSV** | **TỶ LỆ ĐÓNG GÓP** |
| 1 | Đinh Đức Nguyên Vũ | 20128171 | 100% |
| 2 | Nguyễn Tấn Sương | 21133078 | 100% |
| 3 | Bùi Thị Huỳnh Hân | 21133029 | 100% |
| 4 | Phan Khải Huyền | 21133041 | 100% |

**Nhận xét của giảng viên:**

**……………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………………**

*Ngày … tháng 05 năm 2024*

*Giảng viên chấm điểm*

*Ths. Nguyễn Văn Thành*

**LỜI CẢM ƠN**

Chúng em xin gửi lời cảm ơn chân thành đến **trường Đại học Sư phạm Kỹ thuật TPHCM** đã bổ sung môn học Phân Tích Dữ Liệu vào chương trình giảng dạy. Đặc biệt, chúng em xin bày tỏ lòng biết ơn sâu sắc đến giảng viên bộ môn – Thạc sĩ Nguyễn Văn Thành, đã dạy dỗ và truyền đạt những kiến thức quý báu cho chúng em trong suốt thời gian học tập vừa qua. Trong quá trình tham gia lớp học Phân Tích Dữ Liệu của Thầy, chúng em đã thu thập được nhiều kiến thức hữu ích, và đã phát triển tinh thần học tập hiệu quả và nghiêm túc. Những kiến thức này chắc chắn sẽ đóng vai trò quan trọng và trở thành hành trang để chúng em vững bước trong tương lai.Môn học Phân Tích Dữ Liệu thực sự thú vị, mang lại lợi ích to lớn và có tính ứng dụng cao. Nó đảm bảo cung cấp đầy đủ kiến thức, liên kết với nhu cầu thực tế của sinh viên. Mặc dù chúng em đã cố gắng hết sức, nhưng chắc chắn rằng bài đồ án này khó tránh khỏi những thiếu sót và một số vấn đề chưa chính xác. Kính mong Thầy xem xét và đóng góp ý kiến để bài đồ án của chúng em được hoàn thiện hơn.

Nhóm chúng em xin chân thành biết ơn!

Mục Lục

[CHƯƠNG 1: GIỚI THIỆU VỀ ĐỀ TÀI 6](#_Toc167396325)

[1.1.Lý do chọn đề tài 6](#_Toc167396326)

[1.2.Tổng quan về tập dữ liệu 6](#_Toc167396327)

[1.2.1.Nguồn dữ liệu 6](#_Toc167396328)

[1.2.2.Mô tả chi tiết tập dữ liệu 6](#_Toc167396329)

[1.3.Các kiến thức liên quan 6](#_Toc167396330)

[1.3.1. Giới thiệu về mạng nơ ron thần kinh 6](#_Toc167396331)

[1.3.2. Giới thiệu về LSTM 7](#_Toc167396332)

[1.3.3. Ứng dụng LSTM trong dự đoán 7](#_Toc167396333)

[CHƯƠNG 2:PHÂN TÍCH DỮ LIỆU 8](#_Toc167396334)

[2.1.Preprocessing 8](#_Toc167396335)

[2.2.Trực quan trên python 8](#_Toc167396336)

[2.2.1.Import các thư viện cần thiết 8](#_Toc167396337)

[2.2.2.Đọc file CSV 8](#_Toc167396338)

[2.2.3.Kiểm tra tệp dữ liệu 9](#_Toc167396339)

[2.2.4. Trực quan hóa dữ liệu 10](#_Toc167396340)

[2.3.Trực quan trên PowerBI 26](#_Toc167396341)

[2.3.1 Tập dữ liệu facebook.csv 26](#_Toc167396342)

[2.3.2 Tập dữ liệu amazon.csv 27](#_Toc167396343)

[2.3.3 Tập dữ liệu Apple.csv 28](#_Toc167396344)

[2.3.4 Tập dữ liệu Google.csv 29](#_Toc167396345)

[CHƯƠNG 3: DỰ ĐOÁN DỮ LIỆU 30](#_Toc167396346)

[3.1.Sử dụng LTSM và RNN dự đoán tập dữ liệu Google 30](#_Toc167396347)

[3.1.1.Import các thư viện cần thiết 30](#_Toc167396348)

[3.1.2.Đọc file CSV và Đổi kiểu dữ liệu ngày 30](#_Toc167396349)

[3.1.3.Kiểm tra tệp dữ liệu 30](#_Toc167396350)

[3.1.4.Chuẩn bị tập train và set 31](#_Toc167396351)

[3.1.5.Tạo Model 31](#_Toc167396352)

[3.1.6.Huấn luyện mô hình 34](#_Toc167396353)

[3.1.7.Dự đoán và đánh giá các chỉ số sai số 35](#_Toc167396354)

[3.1.8.Trực quan giá trị dự đoán và thực tế 36](#_Toc167396355)

[3.1.9.Sử dụng RNN dự đoán mô hình 36](#_Toc167396356)

[3.1.10.So sánh đánh giá 2 mô hình 39](#_Toc167396357)

[3.2.Sử dụng LTSM và Simple Dense Layer dự đoán tập dữ liệu Apple 39](#_Toc167396358)

[3.2.1.Import các thư viện cần thiết 39](#_Toc167396359)

[3.2.2.Đọc file CSV và In 5 dòng đầu 39](#_Toc167396360)

[3.2.3.Kiểm tra tệp dữ liệu 40](#_Toc167396361)

[3.2.4.Chuẩn bị tập train và set 40](#_Toc167396362)

[3.2.5.Tạo Model 43](#_Toc167396363)

[3.2.6.Huấn luyện mô hình 45](#_Toc167396364)

[3.2.7.Dự đoán 46](#_Toc167396365)

[3.2.8. Trực quan giá trị dự đoán và thực tế 46](#_Toc167396366)

[3.2.9. Sử dụng mô hình Simple Dense Layer để so sánh hiệu suất đánh giá của LSTM 47](#_Toc167396367)

[3.2.10. So sánh 2 mô hình áp dụng và đưa ra kết luận 50](#_Toc167396368)

[CHƯƠNG 4: TỔNG KẾT 51](#_Toc167396369)

[CHƯƠNG 5:TÀI LIỆU THAM KHẢO 51](#_Toc167396370)

BẢNG PHÂN CÔNG NHIỆM VỤ NHÓM 1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Nhiệm Vụ** | **Vũ** | **Huyền** | **Sương** | **Hân** |
| Tìm kiếm tập dữ liệu | X | X | X | X |
| Tiền xử lý,trực quan trên python |  | X |  |  |
| Trực quan trên PowerBI |  |  |  | X |
| Sử dụng RNN dự đoán tập dữ liệu Google | X |  |  |  |
| Sử dụng RNN dữ đoán tập dữ liệu Apple |  |  | X |  |
| Thực hiện ppt | X | X | X | X |
| Thực hiện word | X | X | X | X |

# CHƯƠNG 1: GIỚI THIỆU VỀ ĐỀ TÀI

## 1.1.Lý do chọn đề tài

* Thị trường chứng khoán là một môi trường đầy thách thức và biến động, với giá cổ phiếu được ảnh hưởng bởi nhiều yếu tố khác nhau như tin tức kinh tế, biến động thị trường toàn cầu và các sự kiện địa phương. Do đó, việc có một phương pháp dự đoán giá cổ phiếu chính xác và hiệu quả là rất quan trọng để giúp nhà đầu tư ra quyết định đúng đắn.
* LSTM (Long Short-Term Memory) là một mô hình mạng nơ-ron thần kinh đặc biệt hiệu quả trong việc xử lý dữ liệu chuỗi thời gian. Khả năng của LSTM trong việc ghi nhớ và học từ dữ liệu lịch sử có thể giúp dự đoán giá cổ phiếu dựa trên các mô hình phức tạp và các yếu tố tương quan.
* Bằng cách áp dụng LSTM vào phân tích giá cổ phiếu, ta có thể nắm bắt được các xu hướng và biến động trong giá cổ phiếu, từ đó đưa ra dự đoán về hướng đi tiềm năng của thị trường. Điều này không chỉ hỗ trợ nhà đầu tư trong việc đưa ra quyết định đầu tư, mà còn mang lại cơ hội cho các nhà nghiên cứu và chuyên gia tài chính trong việc phát triển các chiến lược giao dịch mới và cải thiện hiệu suất đầu tư.

## 1.2.Tổng quan về tập dữ liệu

### 1.2.1.Nguồn dữ liệu

* -Tập dữ liệu sử dụng gồm 4 file csv về giá cổ phiếu của 4 công ty:Amazon,Apple,Facebook,Google
* Tập dữ liệu được lấy từ Kaggle là một bộ dữ liệu chứa thông tin về giá cổ phiếu lịch sử của 10 công ty phổ biến. Dữ liệu này được tổng hợp từ một số nguồn tin cậy và bao gồm thông tin về giá mở cửa (Open), giá cao nhất (High), giá thấp nhất (Low), giá đóng cửa (Close) và khối lượng giao dịch (Volume) của các cổ phiếu trong một khoảng thời gian nhất định.

### 1.2.2.Mô tả chi tiết tập dữ liệu

-Tập dữ liệu gồm 7 cột : Date,Open,High,Low,Adj close,Volumn.

## 1.3.Các kiến thức liên quan

### 1.3.1. Giới thiệu về mạng nơ ron thần kinh

* Mạng nơ ron thần kinh (Neural Networks) là một mô hình tính toán được lấy cảm hứng từ cấu trúc và chức năng của bộ não con người. Các mạng này bao gồm các đơn vị xử lý (neurons) được kết nối với nhau theo nhiều lớp: lớp đầu vào, lớp ẩn, và lớp đầu ra. Chức năng chính của mạng nơ ron thần kinh là học từ dữ liệu để nhận diện các mẫu và đưa ra dự đoán.
* Một mạng nơ ron thần kinh cơ bản hoạt động qua ba giai đoạn:
* Lan truyền tiến (Forward Propagation): Dữ liệu đầu vào được truyền qua các lớp của mạng để tạo ra dự đoán đầu ra.
* Tính toán hàm mất mát (Loss Function): Hàm mất mát đánh giá độ chính xác của dự đoán so với giá trị thực.
* Lan truyền ngược (Backward Propagation): Dựa trên giá trị của hàm mất mát, mạng cập nhật các trọng số của các kết nối bằng cách sử dụng các thuật toán tối ưu hóa như Gradient Descent.

### 1.3.2. Giới thiệu về LSTM

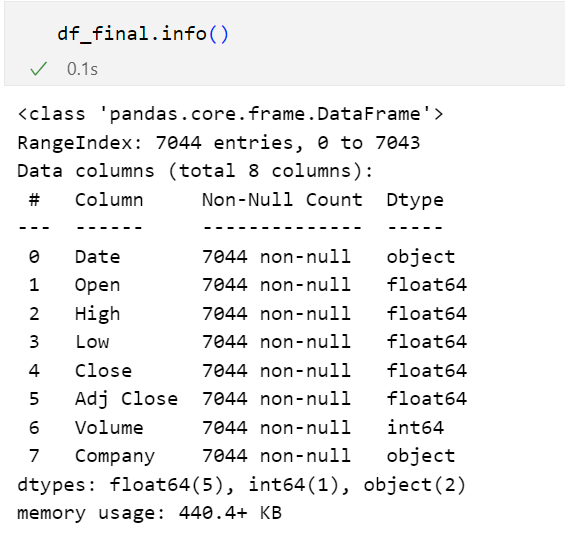
* LSTM (Long Short-Term Memory) là một loại mạng nơ ron hồi quy (Recurrent Neural Network - RNN) được thiết kế để xử lý và dự đoán dữ liệu theo chuỗi thời gian, vượt qua những hạn chế của các RNN thông thường. LSTM đặc biệt hiệu quả trong việc duy trì thông tin trong thời gian dài và xử lý các vấn đề liên quan đến gradient biến mất hoặc bùng nổ.
* Cấu trúc cơ bản của một tế bào LSTM bao gồm:
  + Cổng vào (Input Gate): Quyết định lượng thông tin mới sẽ được thêm vào trạng thái tế bào.
  + Cổng quên (Forget Gate): Quyết định lượng thông tin cần loại bỏ khỏi trạng thái tế bào.
  + Cổng đầu ra (Output Gate): Quyết định lượng thông tin sẽ được xuất ra khỏi tế bào để đưa đến bước tiếp theo.
* Các đặc điểm nổi bật của LSTM:
  + Khả năng ghi nhớ dài hạn: Nhờ cấu trúc các cổng, LSTM có khả năng ghi nhớ và duy trì thông tin qua các bước thời gian dài.
  + Tránh gradient biến mất và bùng nổ: Bằng cách điều chỉnh cách thông tin được lưu trữ và cập nhật, LSTM giúp duy trì sự ổn định của quá trình học.

### 1.3.3. Ứng dụng LSTM trong dự đoán

* LSTM đã được ứng dụng rộng rãi trong nhiều lĩnh vực yêu cầu dự đoán dựa trên dữ liệu chuỗi thời gian. Một số ứng dụng phổ biến bao gồm:
  + Dự báo tài chính: LSTM được sử dụng để dự đoán giá cổ phiếu, tỷ giá hối đoái và các chỉ số tài chính khác.
  + Xử lý ngôn ngữ tự nhiên: Trong các tác vụ như dự đoán từ tiếp theo trong văn bản, phân loại cảm xúc, và dịch máy.
  + Nhận dạng giọng nói: Giúp cải thiện độ chính xác của các hệ thống nhận dạng giọng nói bằng cách xử lý chuỗi âm thanh.
  + Dự báo nhu cầu: LSTM có thể dự đoán nhu cầu của sản phẩm trong tương lai dựa trên dữ liệu bán hàng trong quá khứ.
  + Phân tích chuỗi thời gian: Ứng dụng trong việc dự đoán xu hướng thời tiết, sản xuất năng lượng, và các hiện tượng khác theo thời gian.
  + Nhờ khả năng xử lý các quan hệ dài hạn và tính năng động trong việc quản lý thông tin, LSTM đã trở thành công cụ quan trọng trong việc phân tích và dự đoán dữ liệu chuỗi thời gian phức tạp.

# CHƯƠNG 2:PHÂN TÍCH DỮ LIỆU

## 2.1.Preprocessing



## 2.2.Trực quan trên python

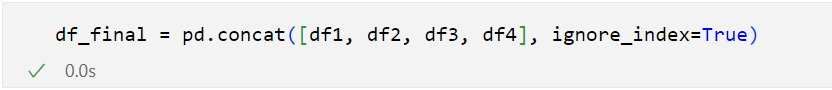
### 2.2.1.Import các thư viện cần thiết



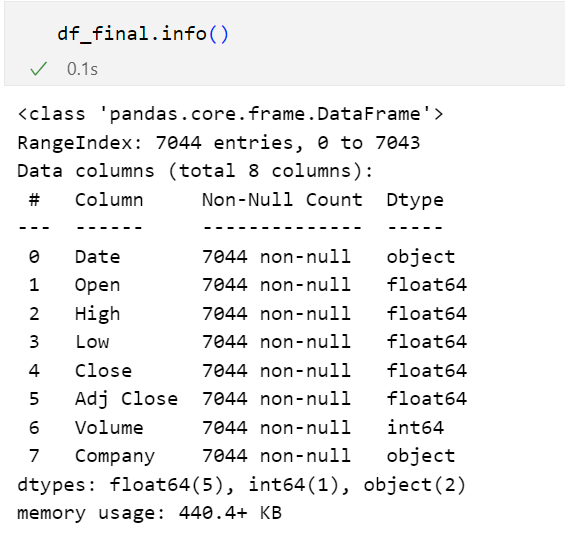
### 2.2.2.Đọc file CSV

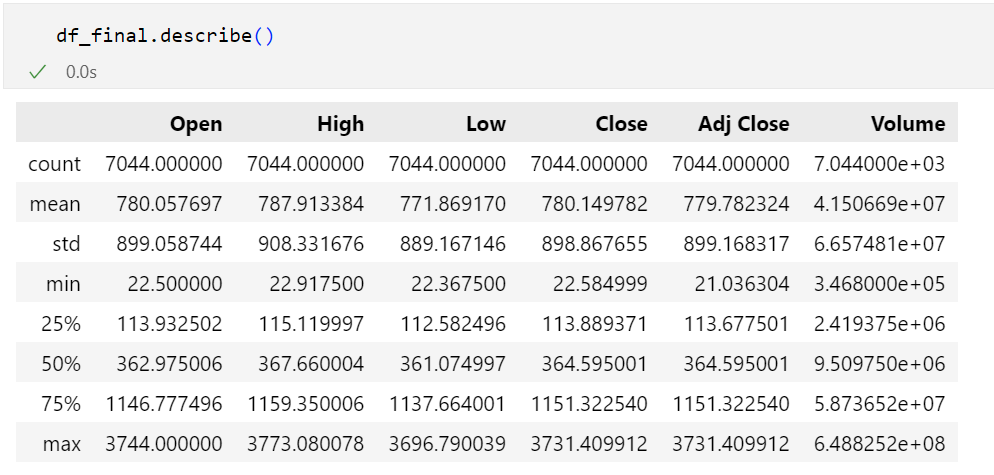


- Sử dụng hàm pd.concat để kết hợp các DataFrame df1, df2, df3, và df4 thành một DataFrame duy nhất df\_final.



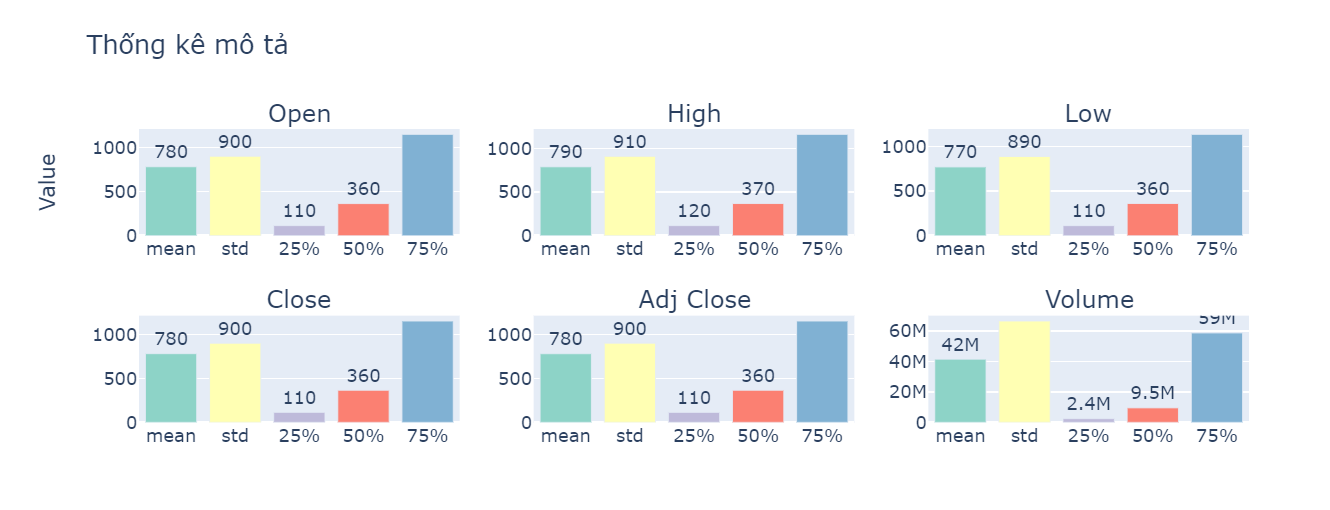
### 2.2.3.Kiểm tra tệp dữ liệu

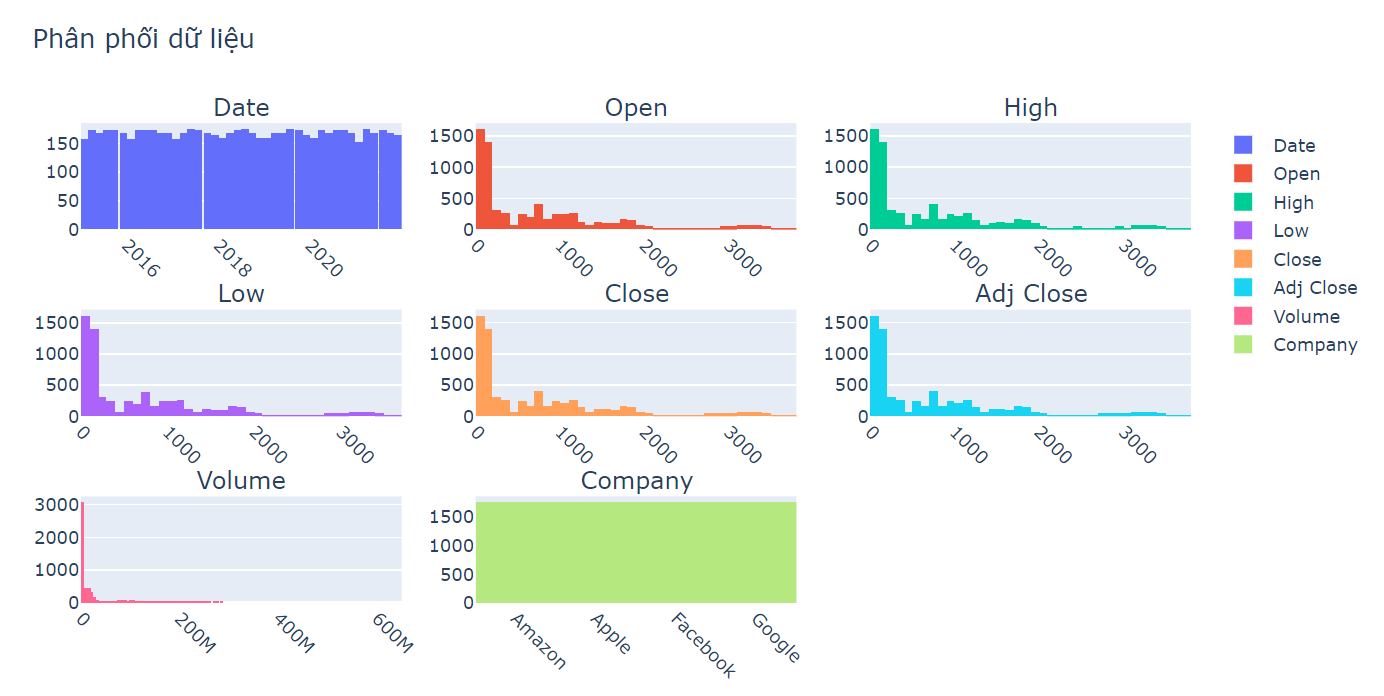




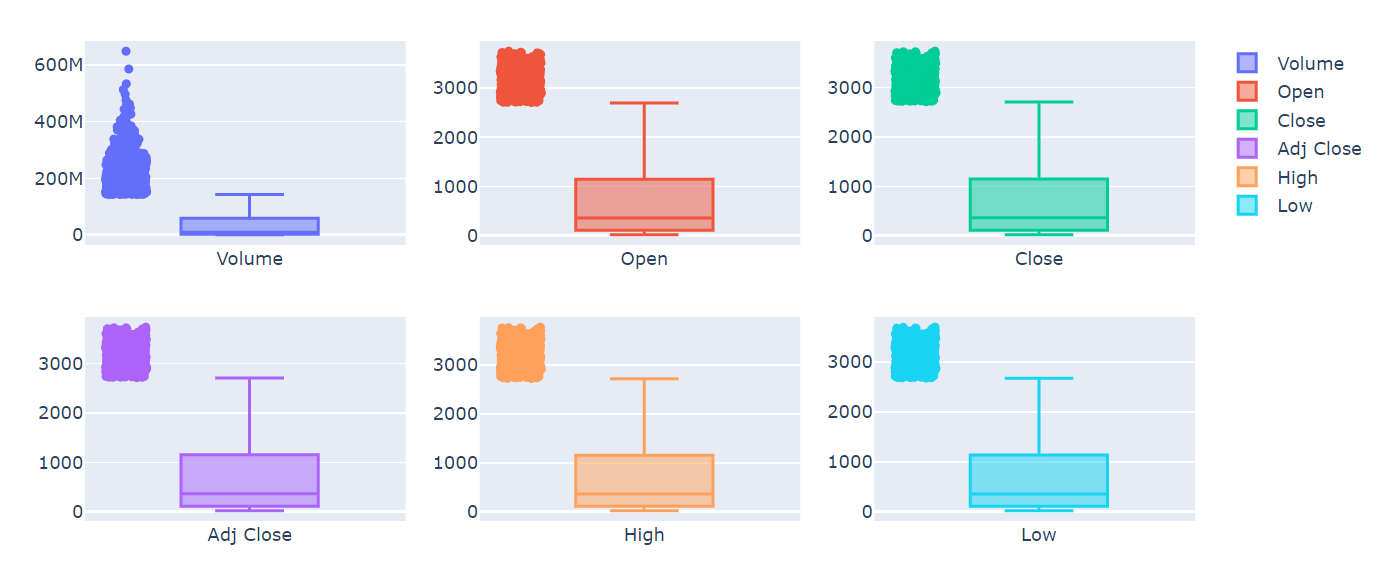
### 2.2.4. Trực quan hóa dữ liệu

a. Phân phối & Thống kê Dữ liệu



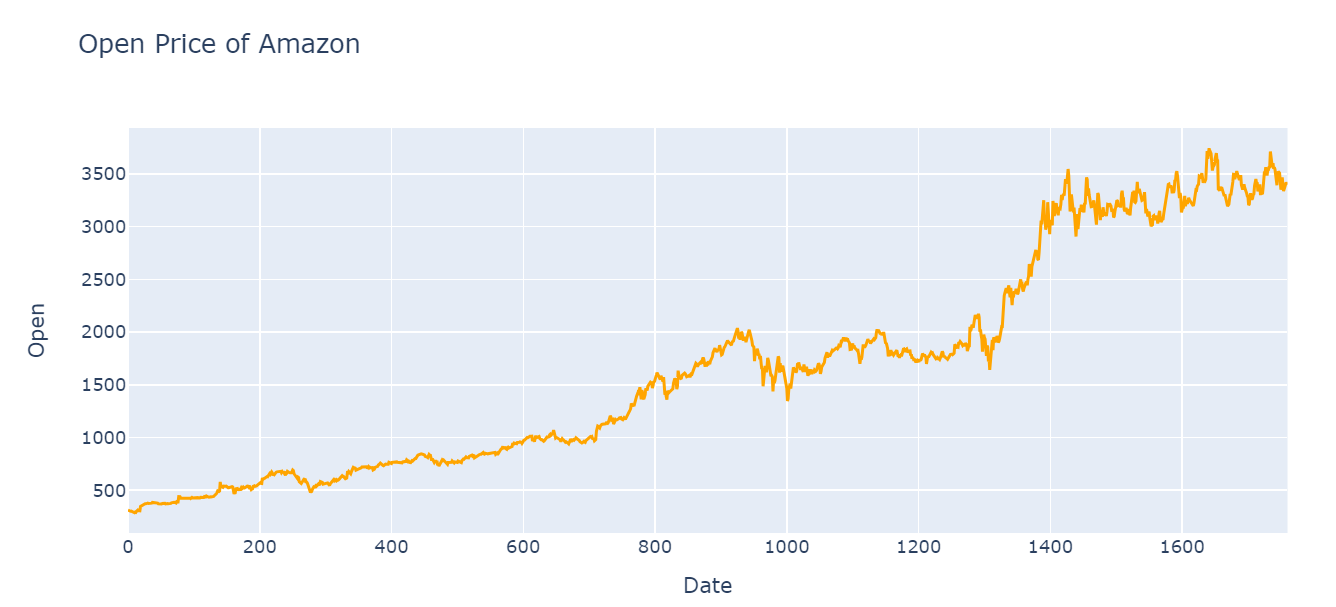


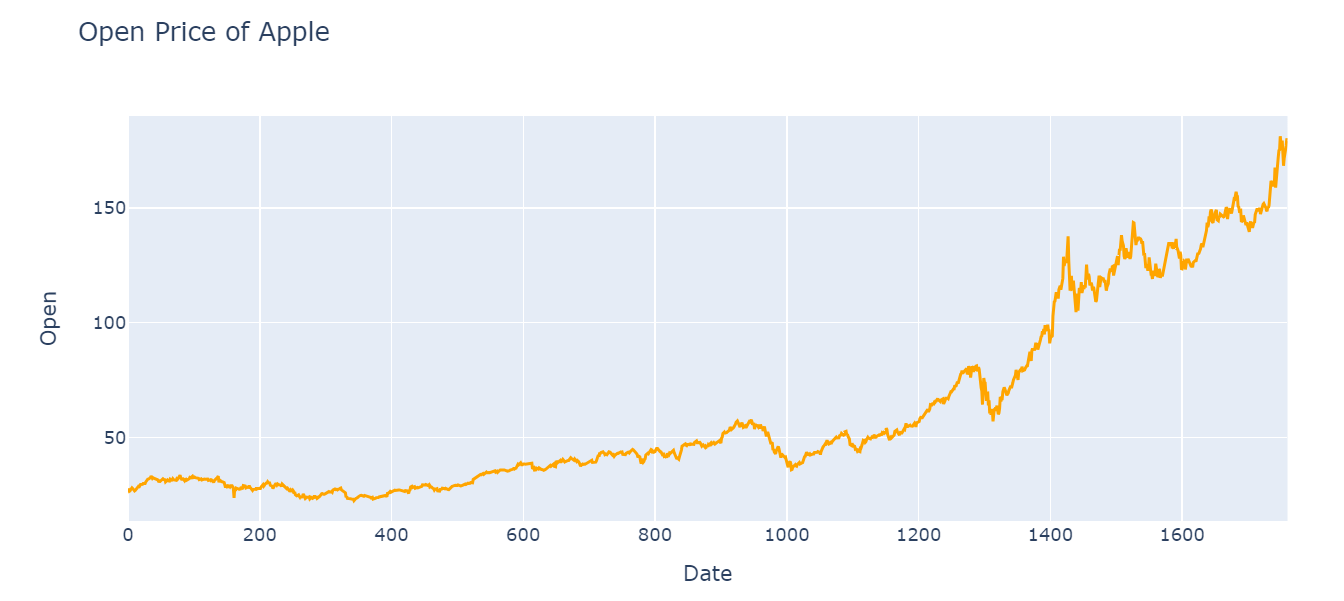
b. Trực quan hóa phân phối dữ liệu và phát hiện các giá trị ngoại lai

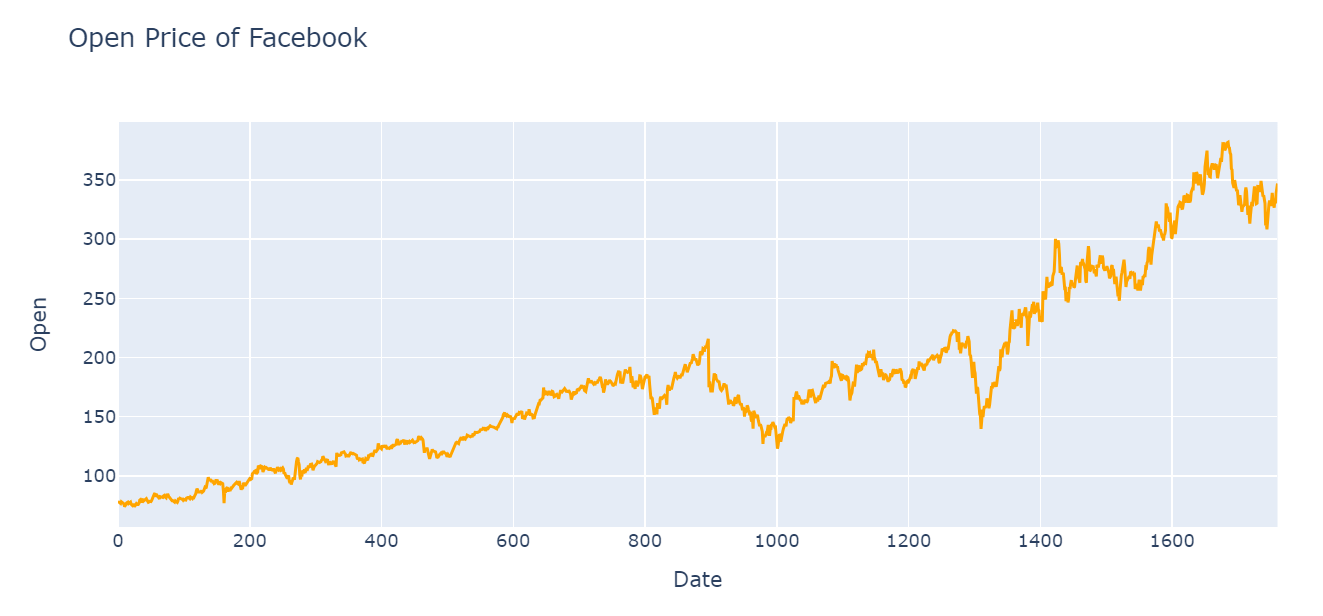


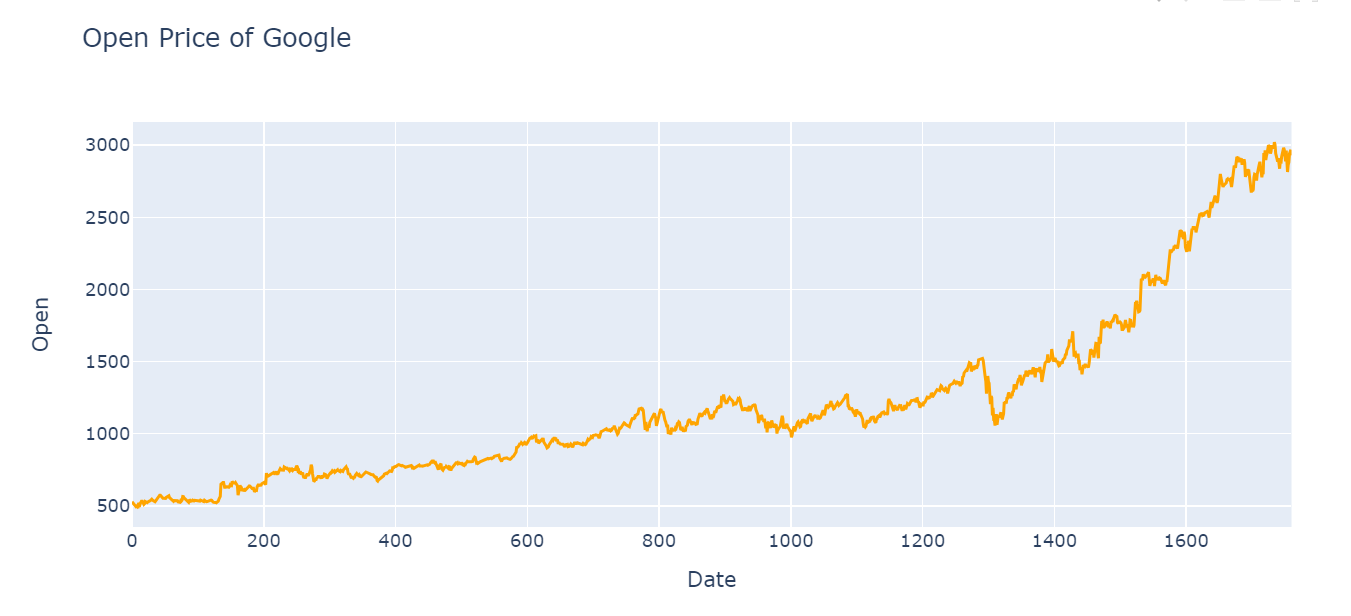
c. Trực quan hóa dữ liệu

\* Phân tích giá cổ phiếu các công ty qua các năm

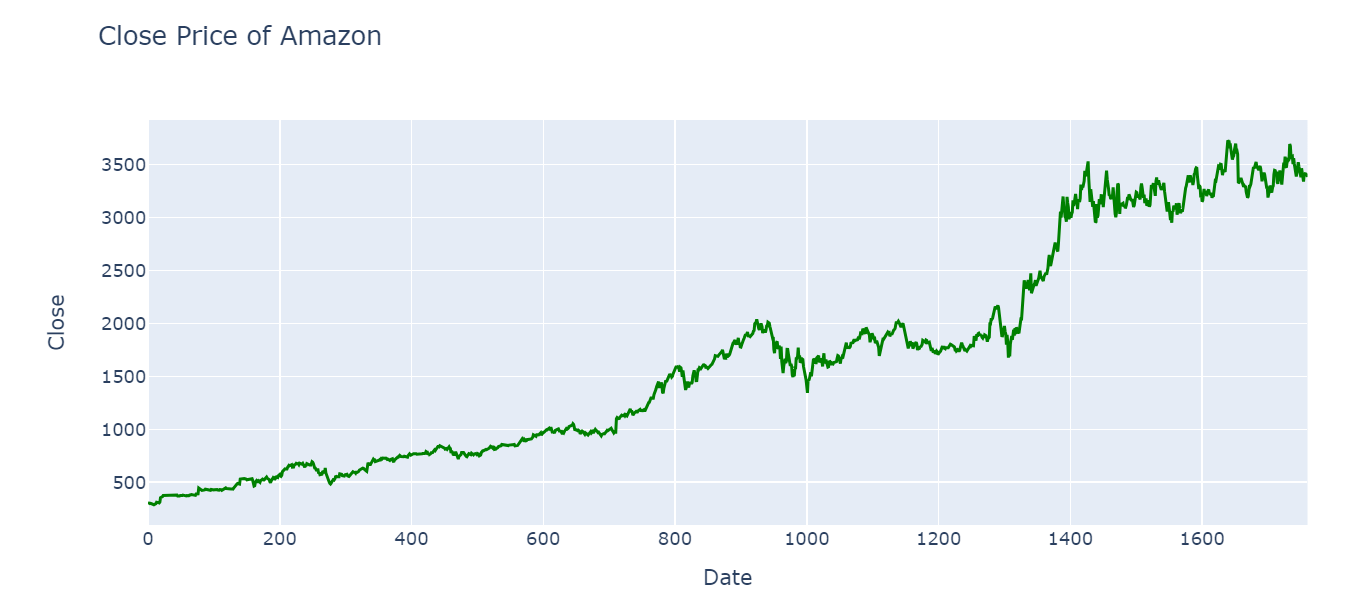
- Open Price  


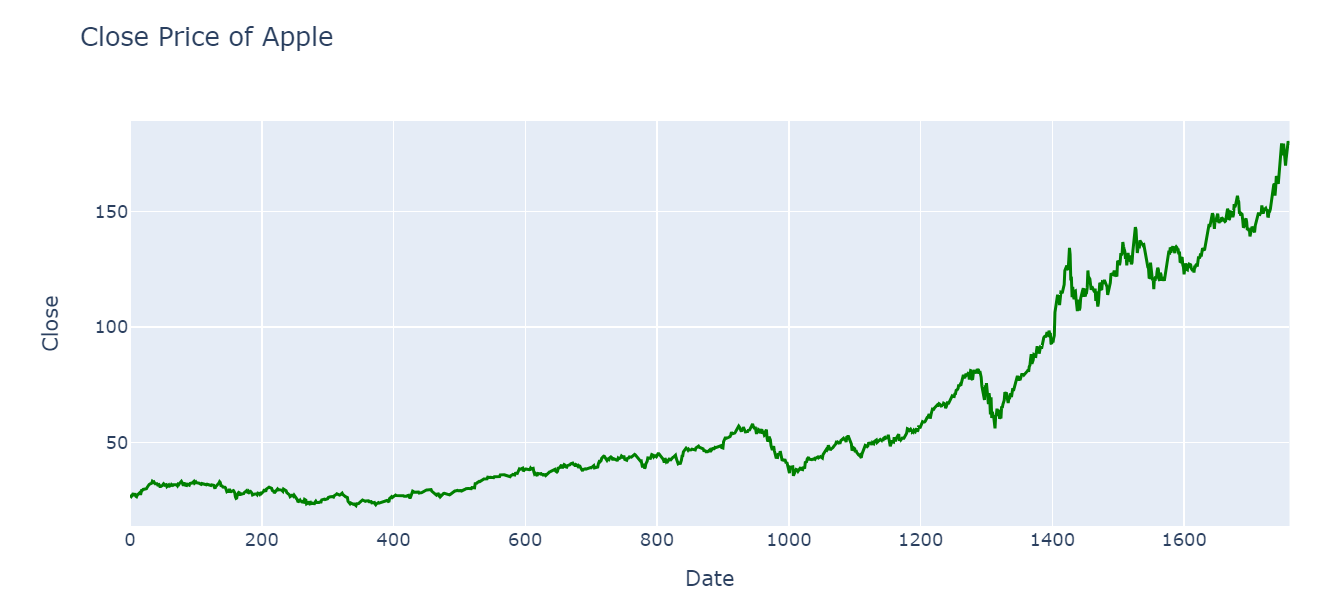


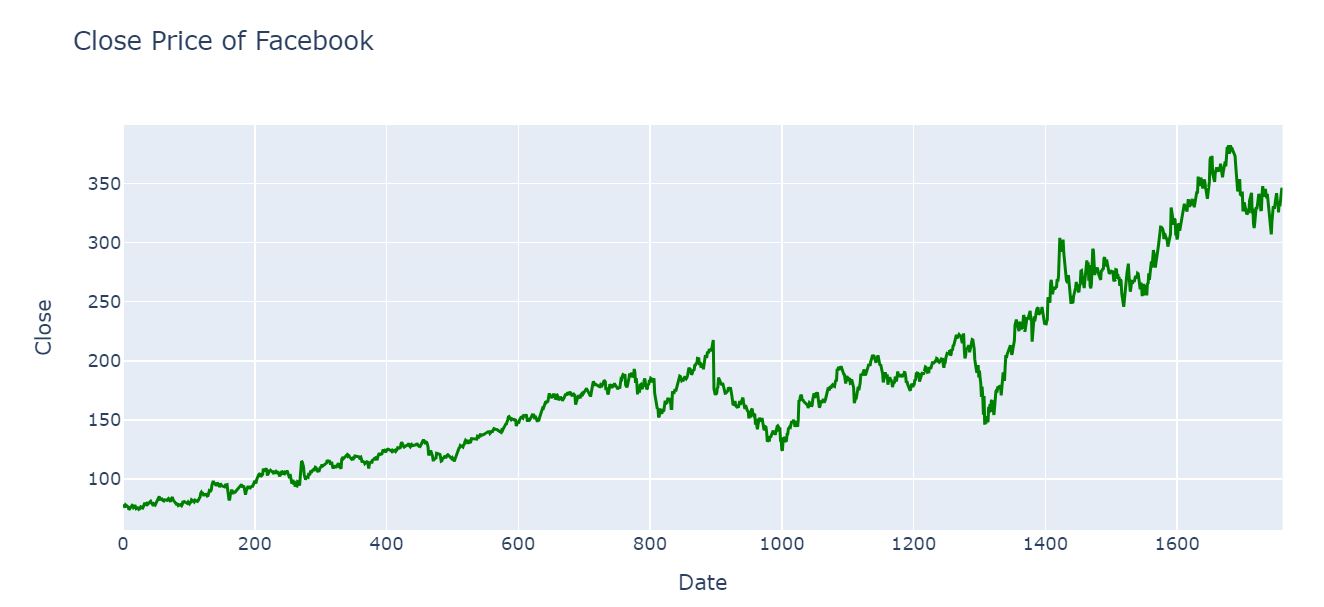


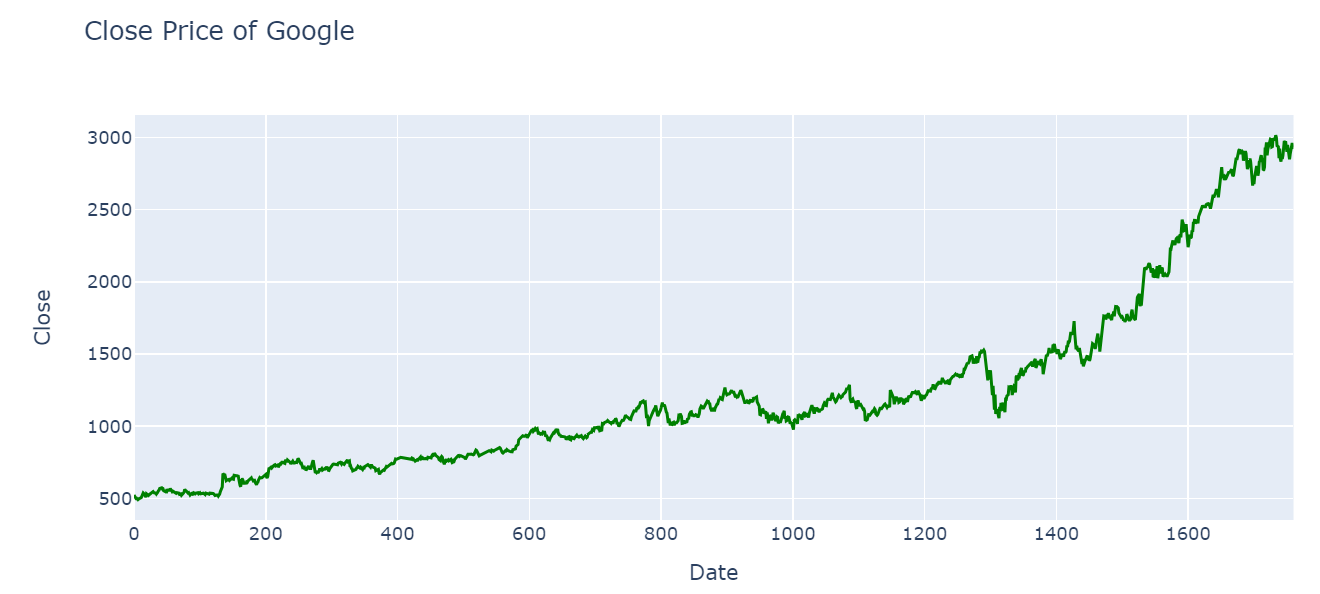


- Close Price

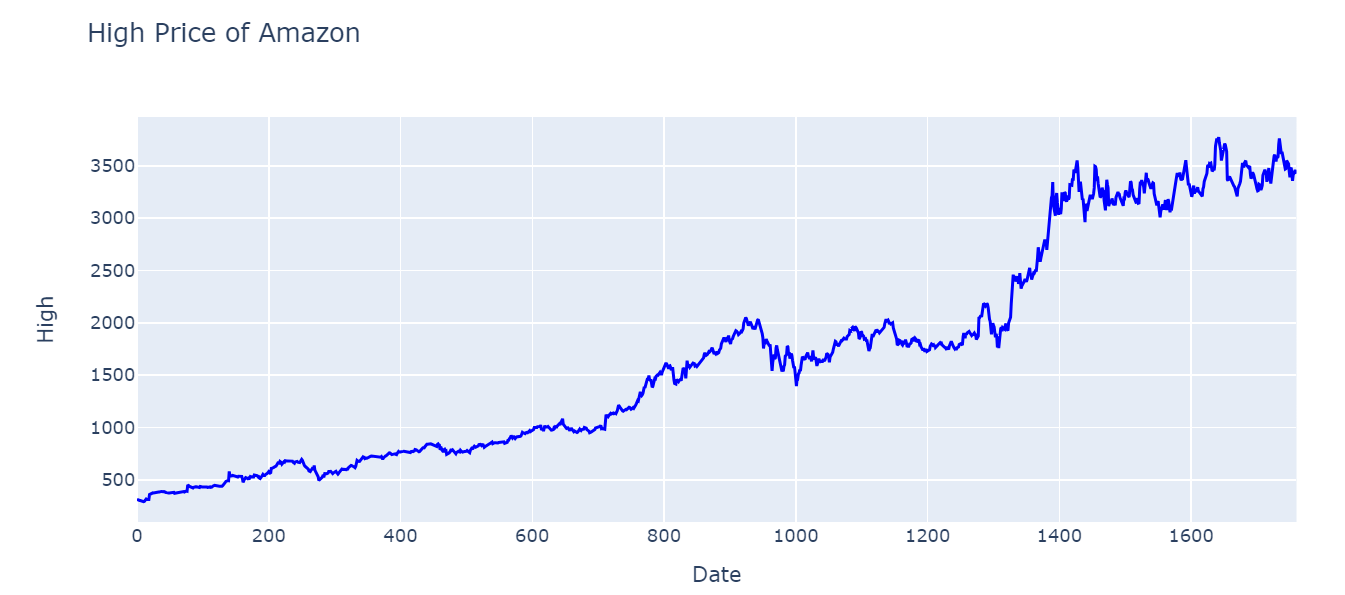


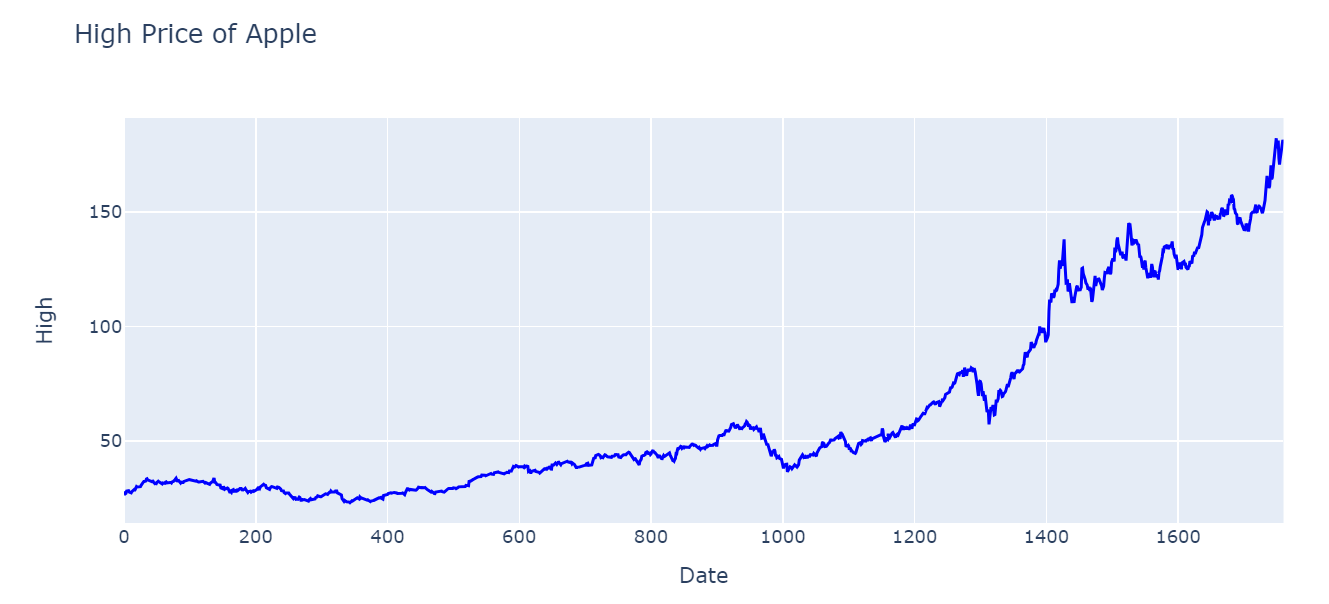


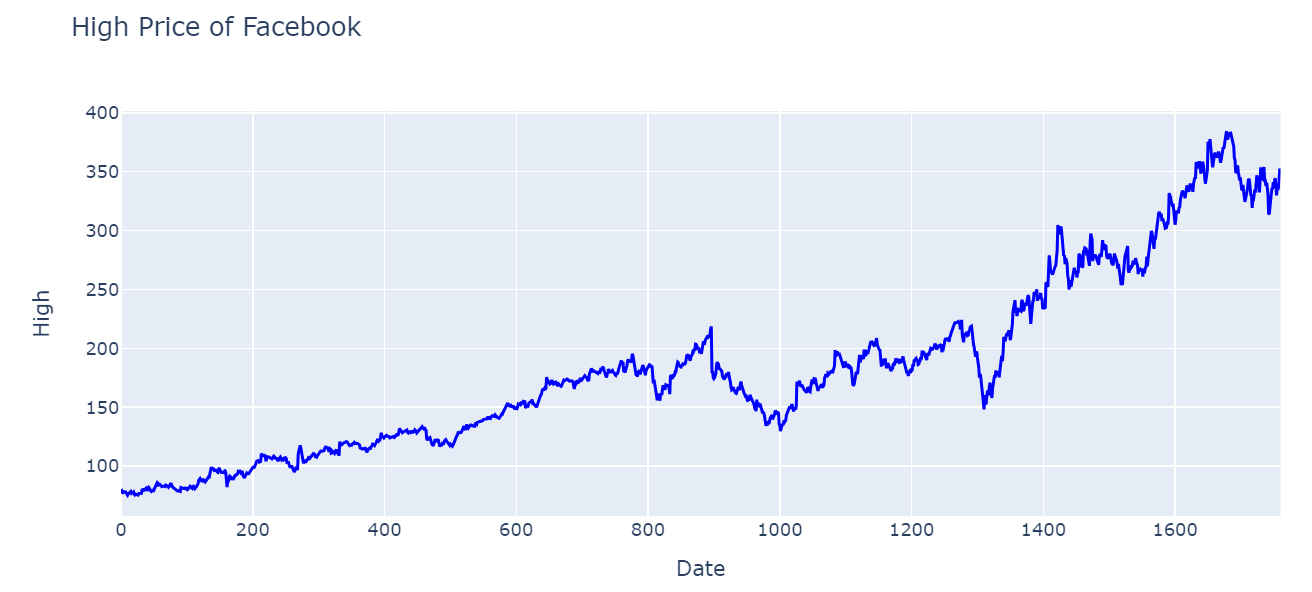


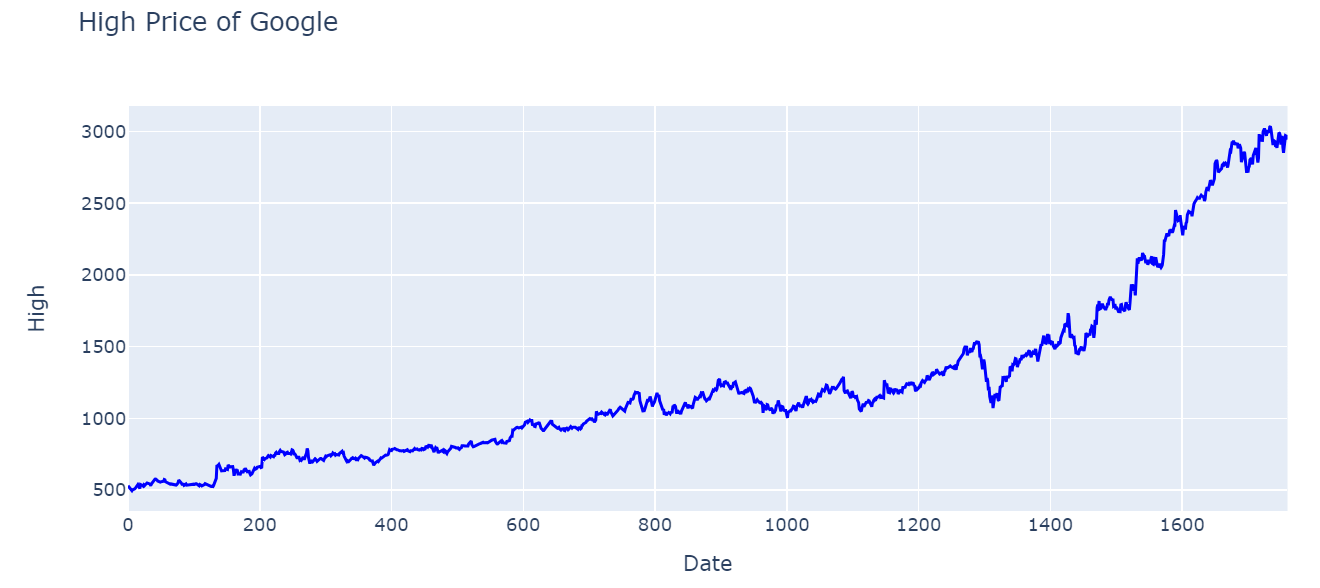


- High Price

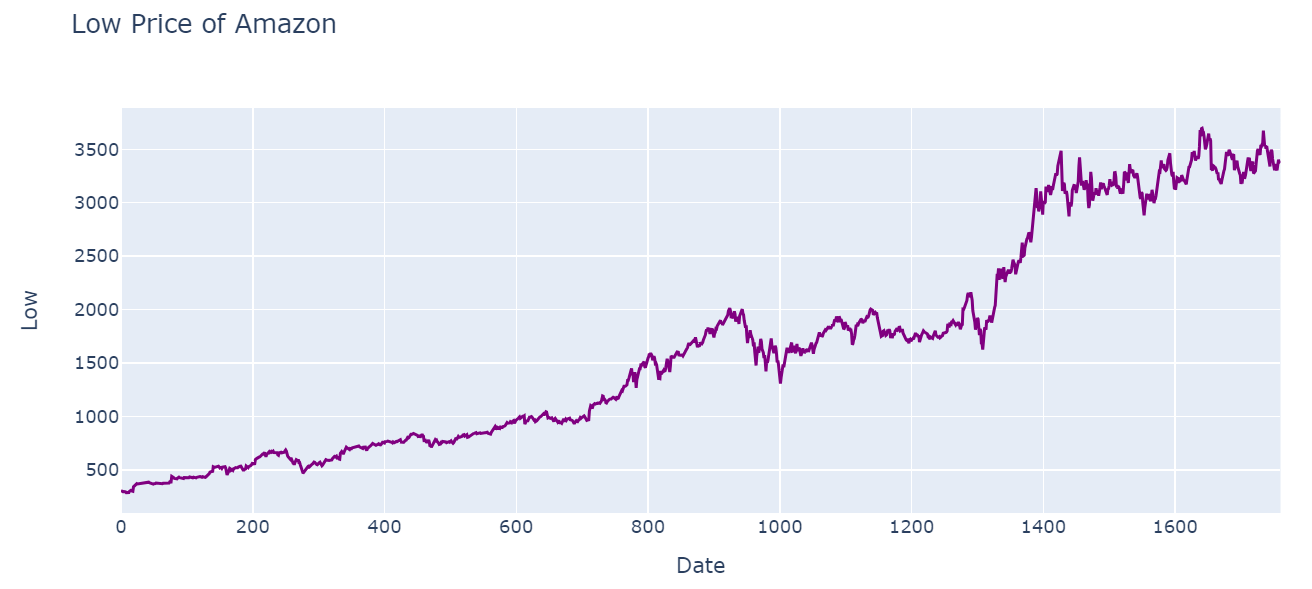


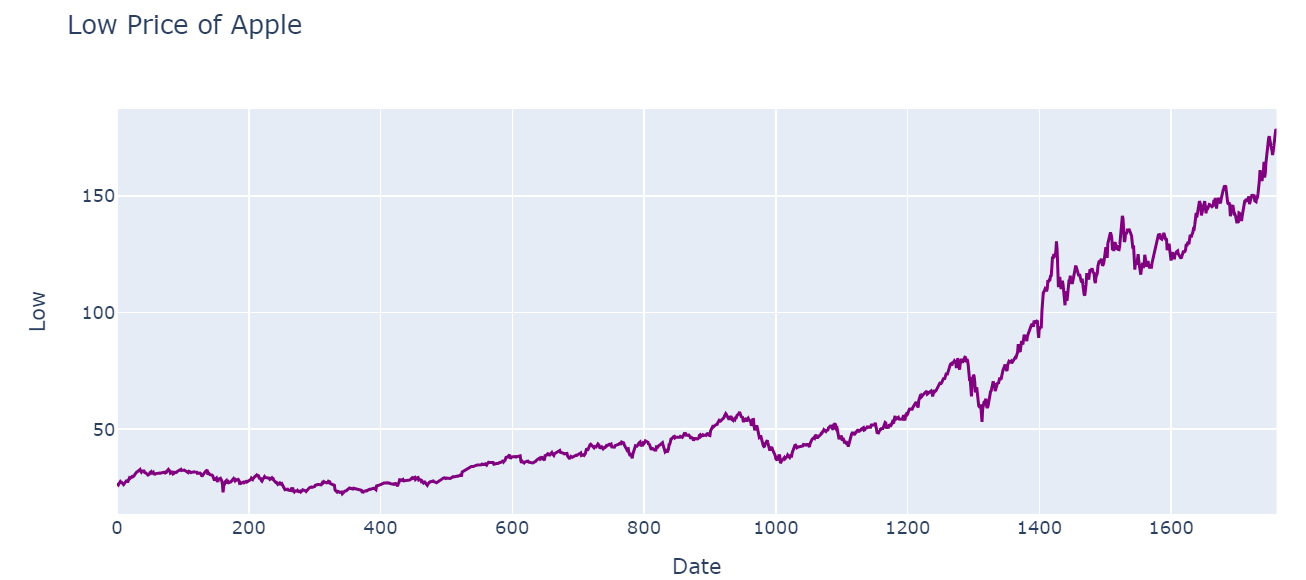


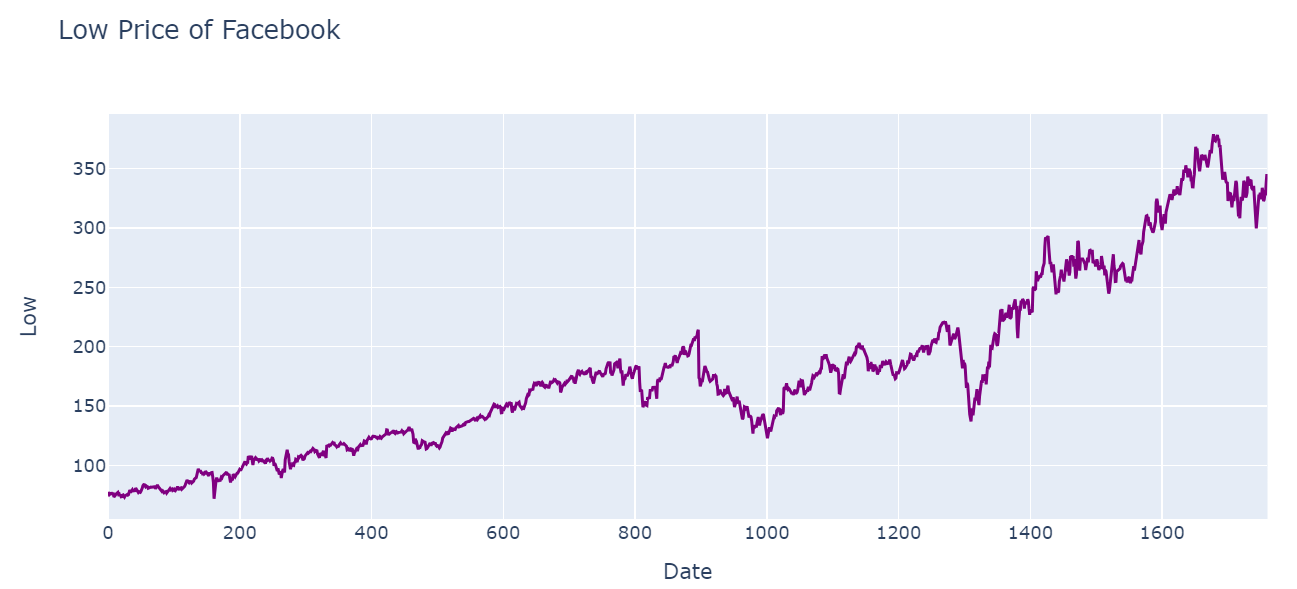


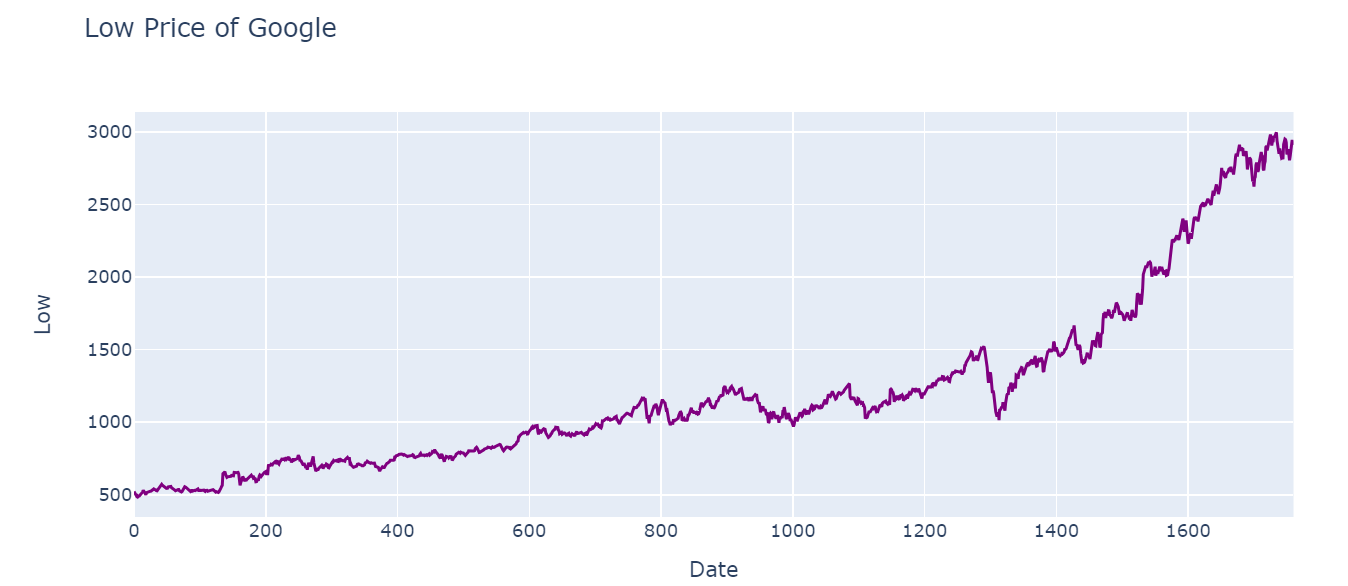


- Low Price

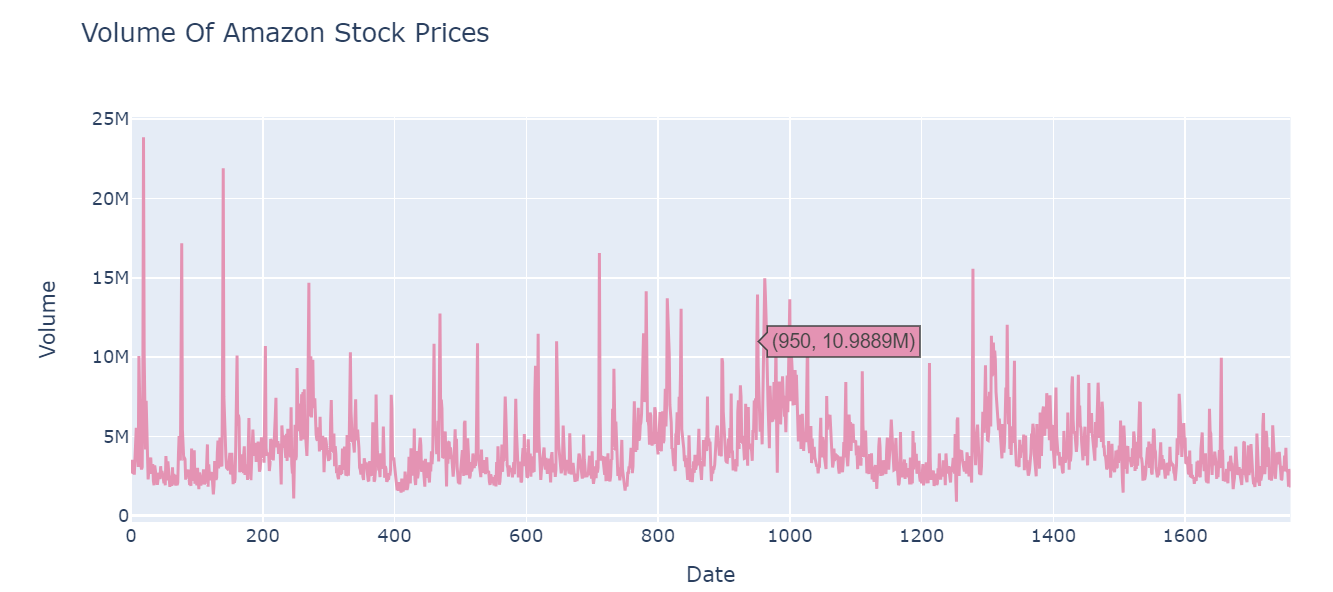


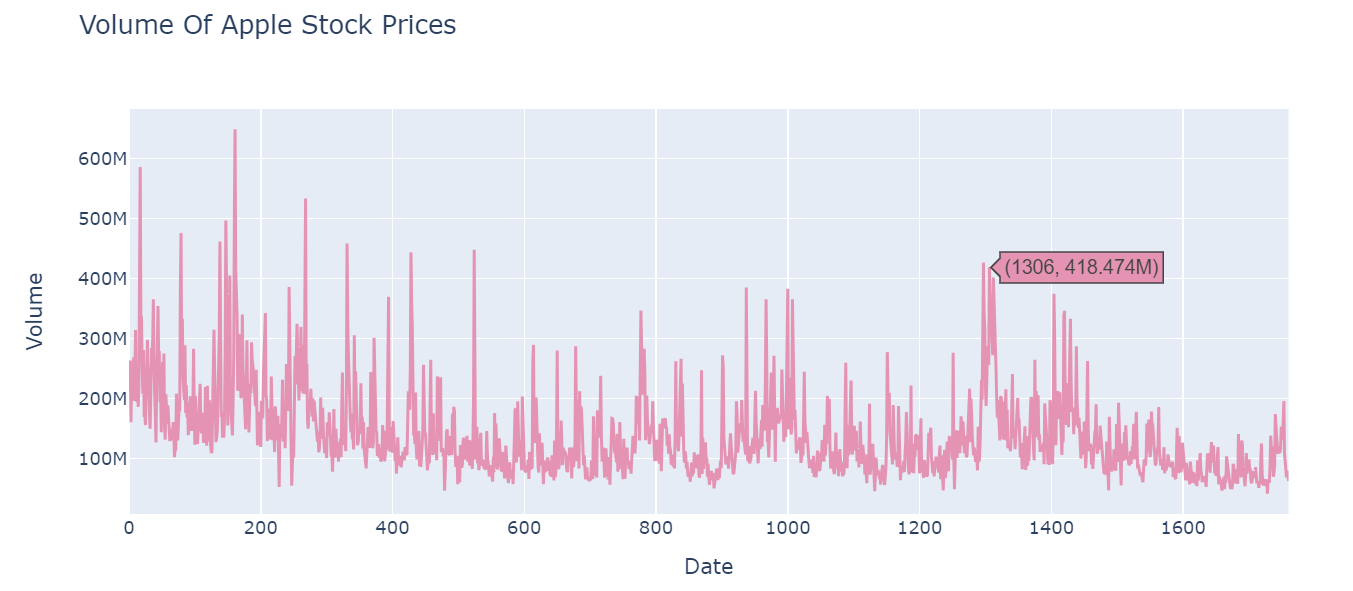




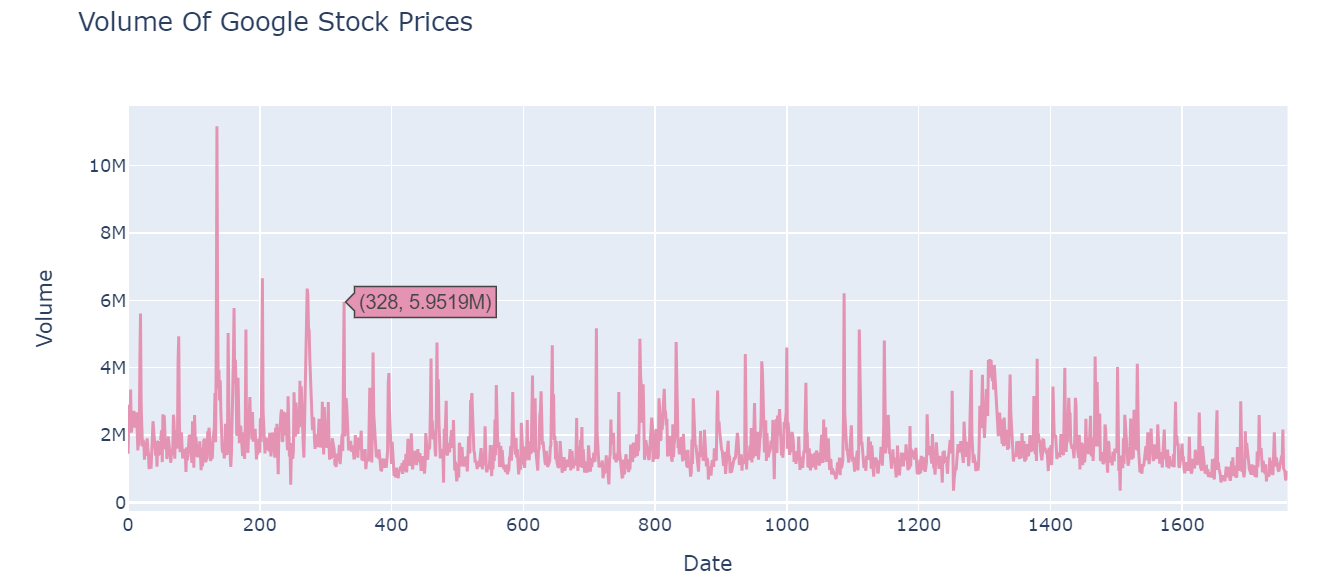


- Volume



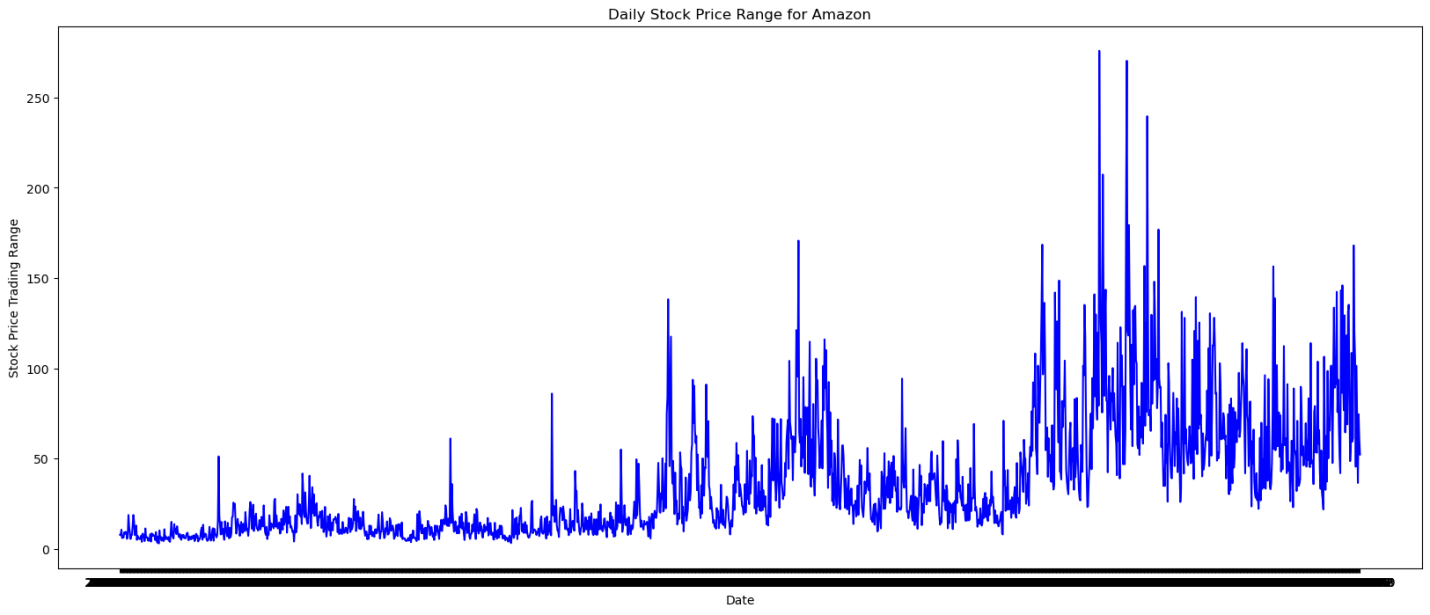


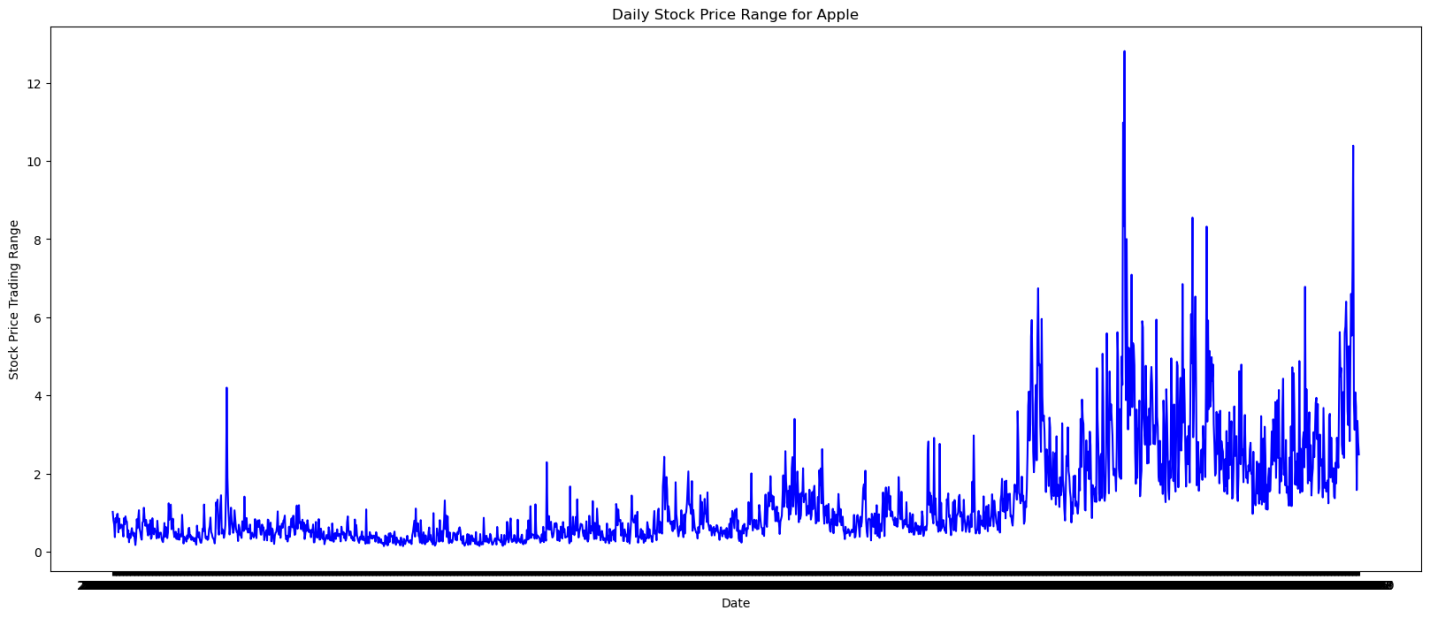


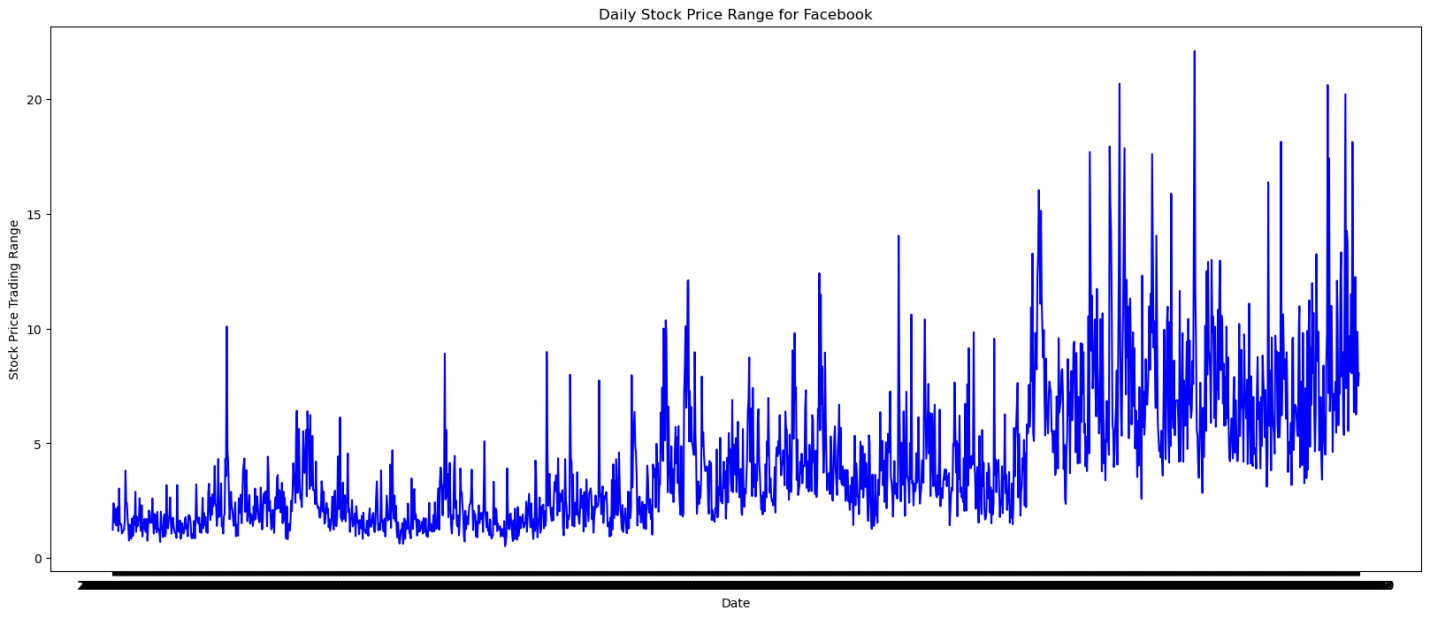


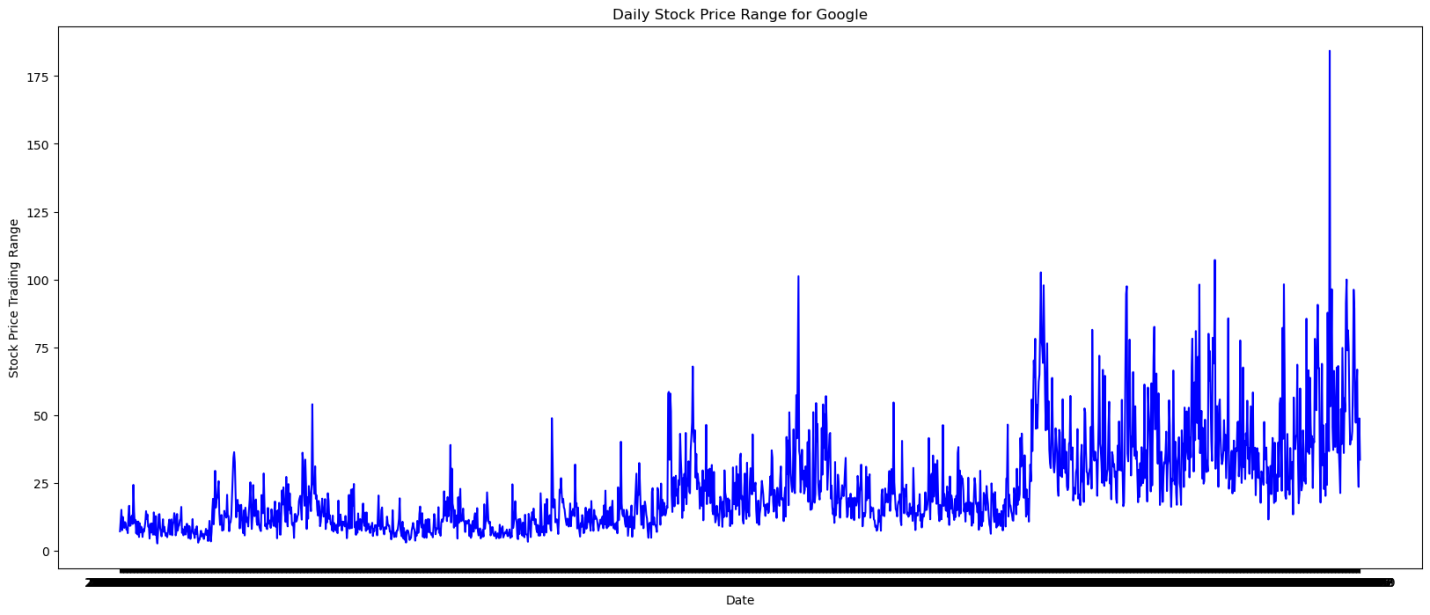
\* Phân tích so sánh giữa giá cao nhất và giá thấp nhất của cổ phiếu qua các năm, cho thấy sự thay đổi trong biên độ dao động hàng ngày của giao dịch.



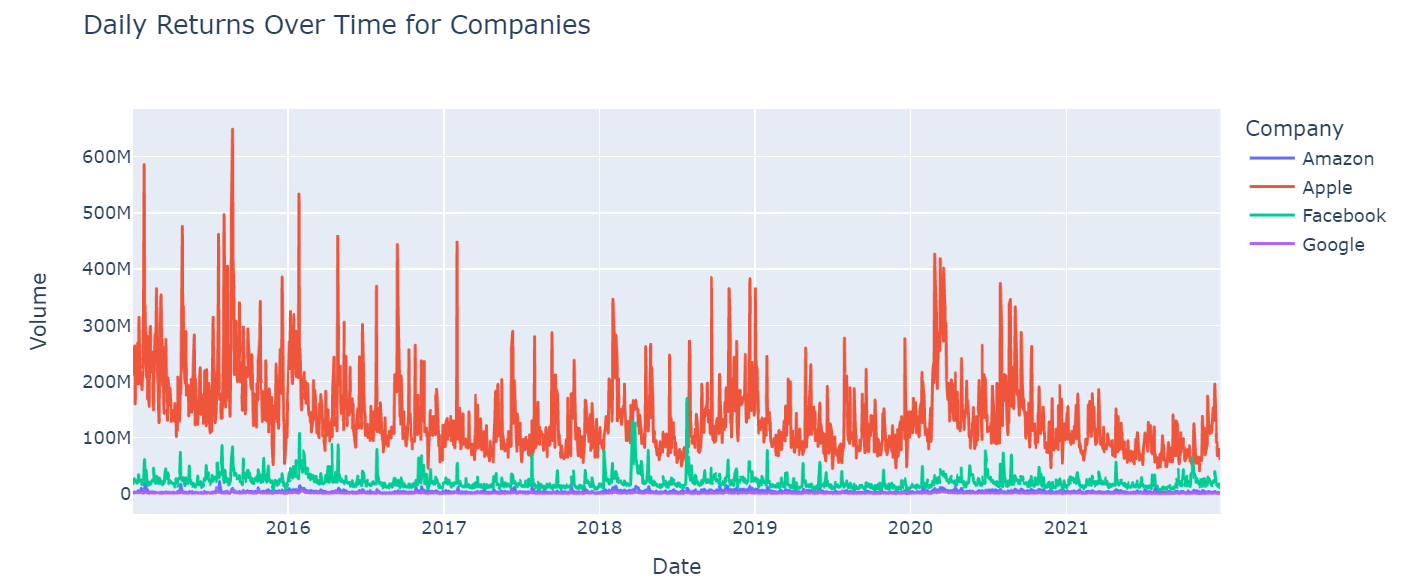




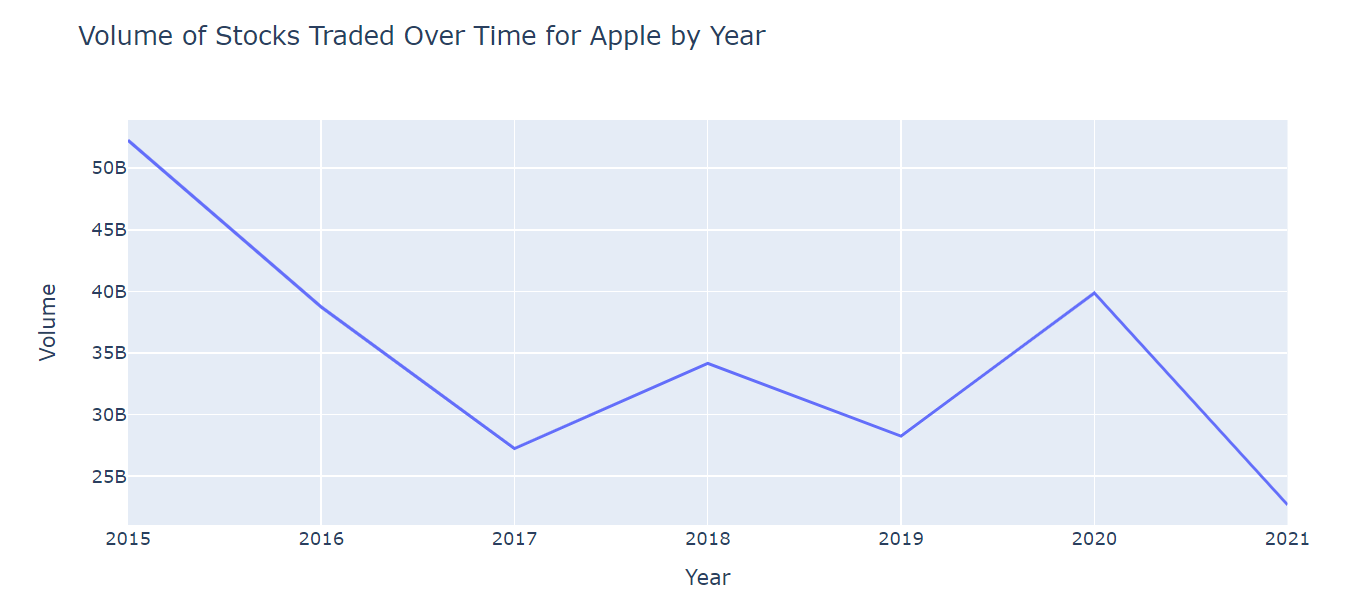


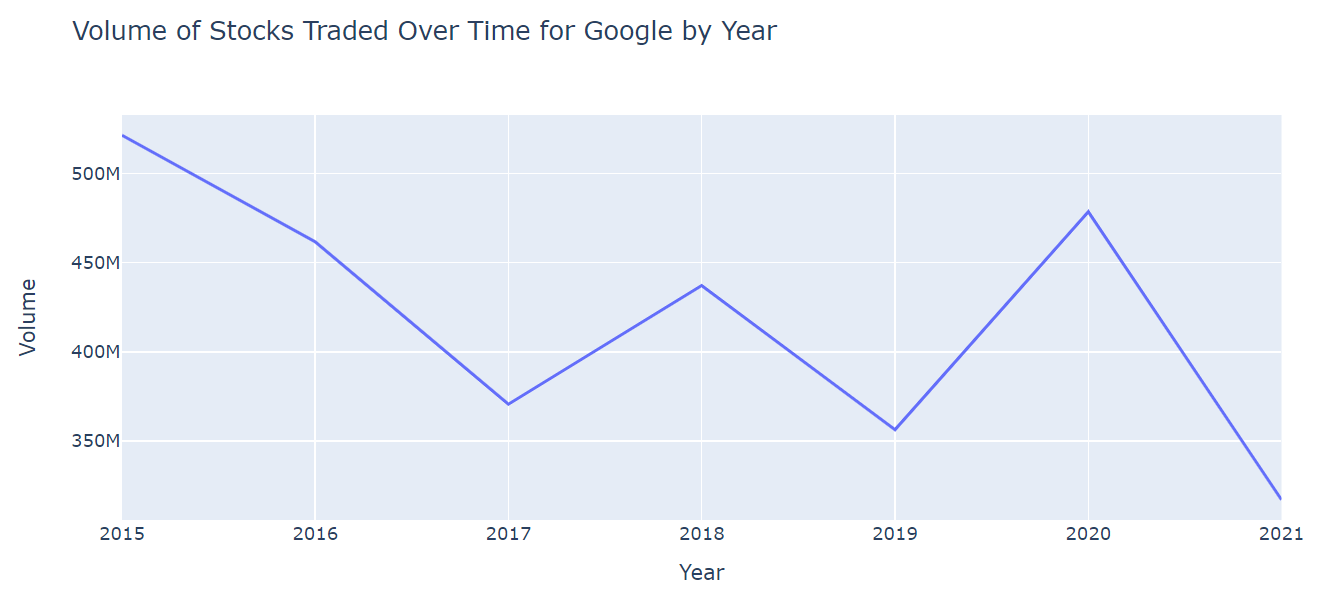
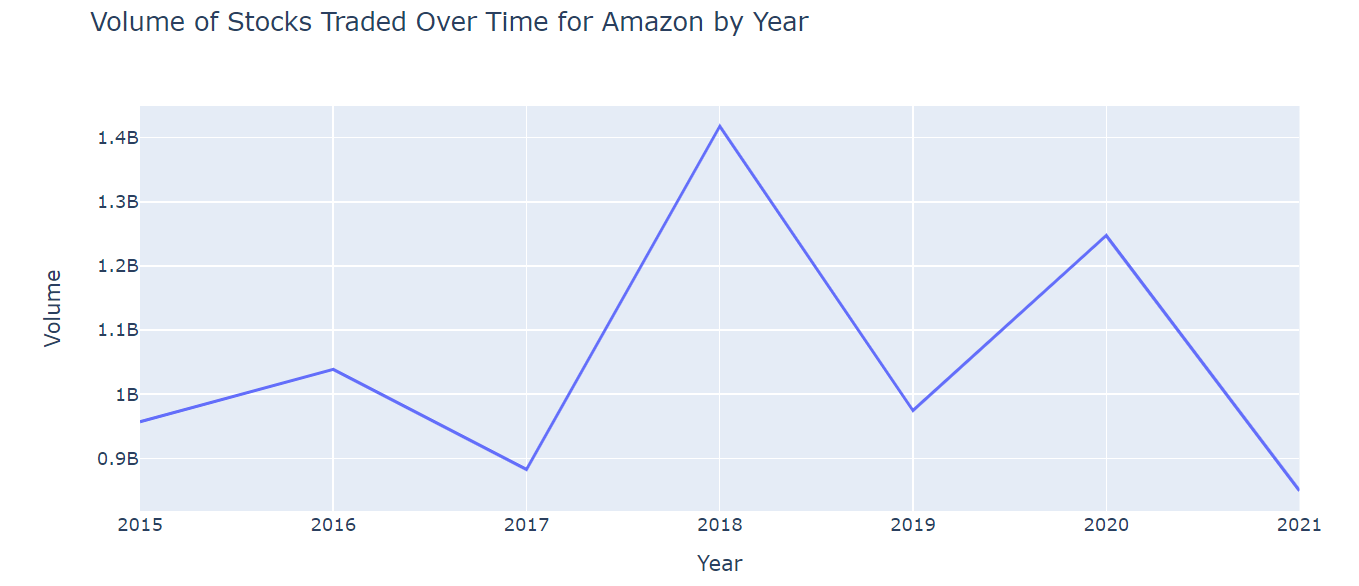


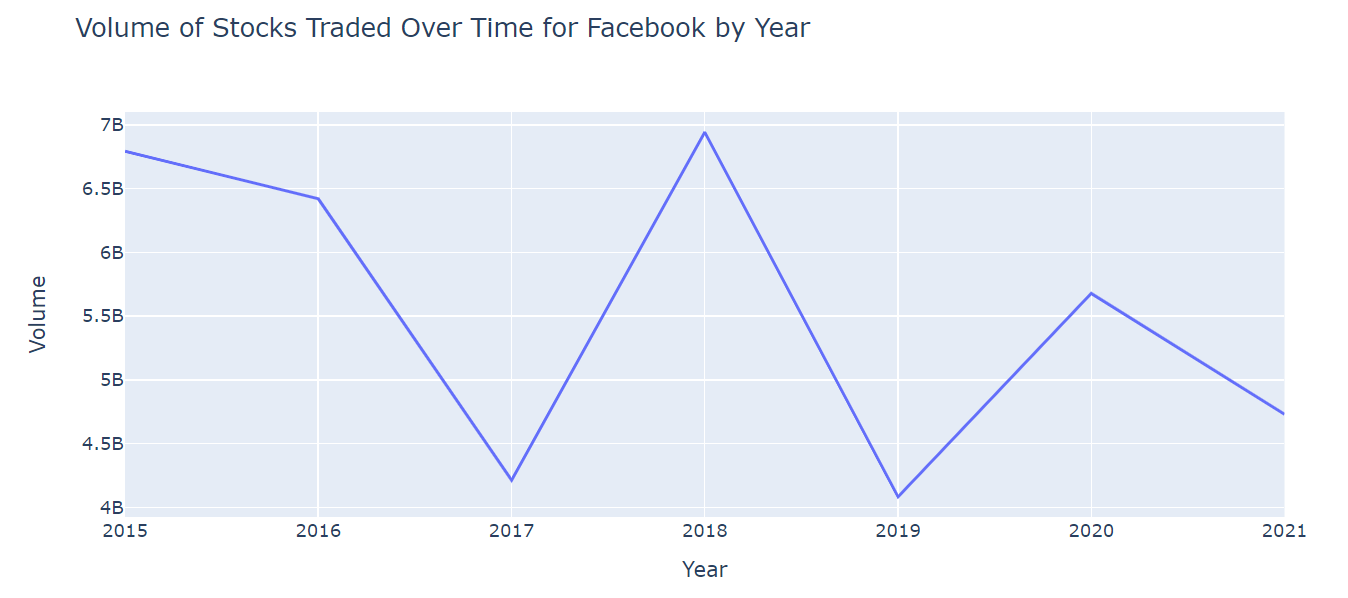
\* Lợi nhuận hằng ngày theo thời gian



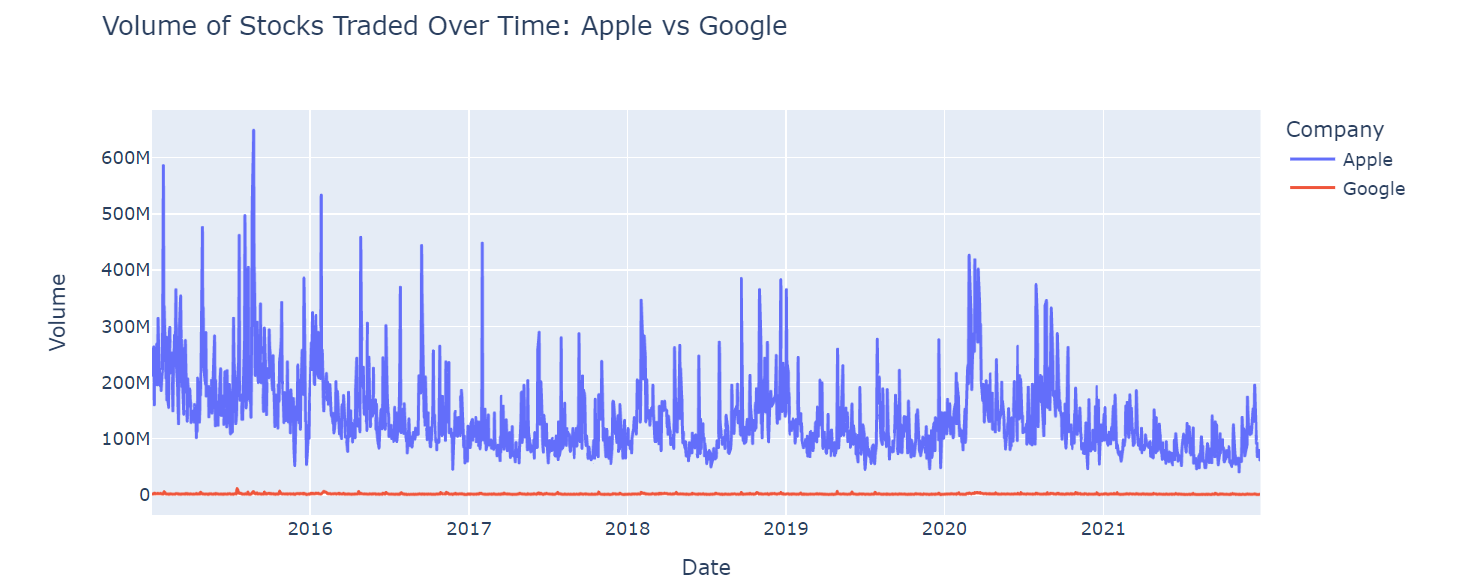
\* Khối lượng cổ phiếu giao dịch theo thời gian



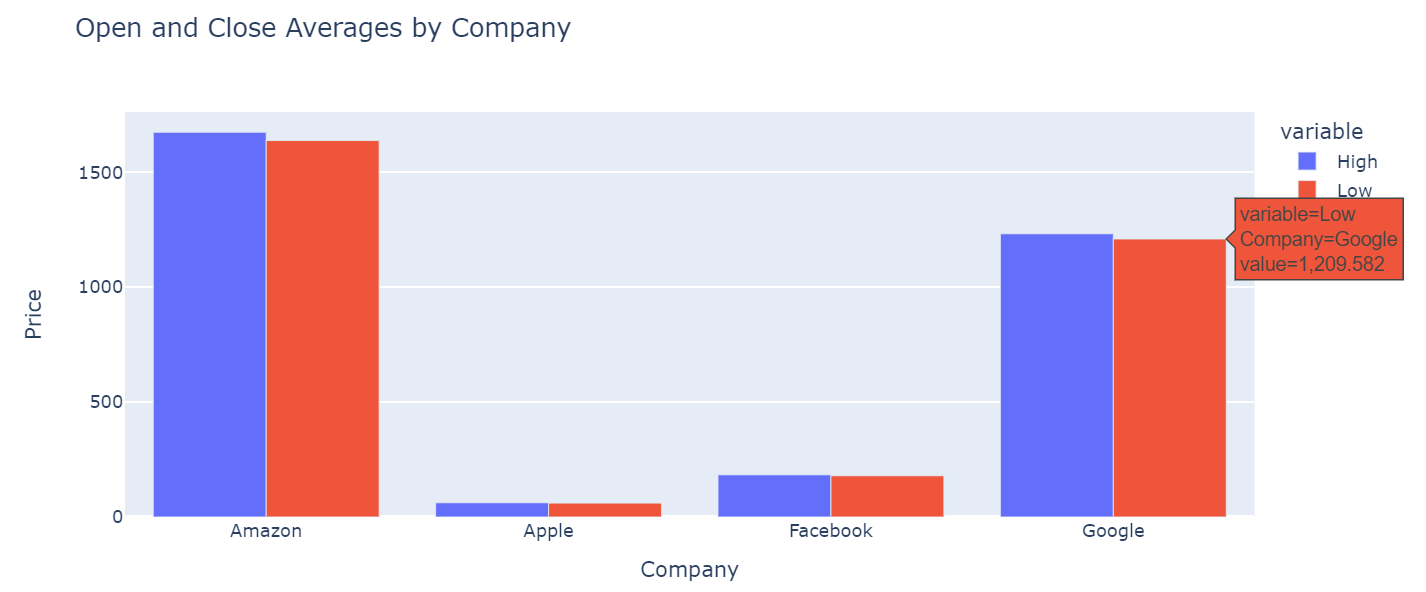
 



\* Khối lượng cổ phiếu giao dịch theo thời gian: Apple so với Google



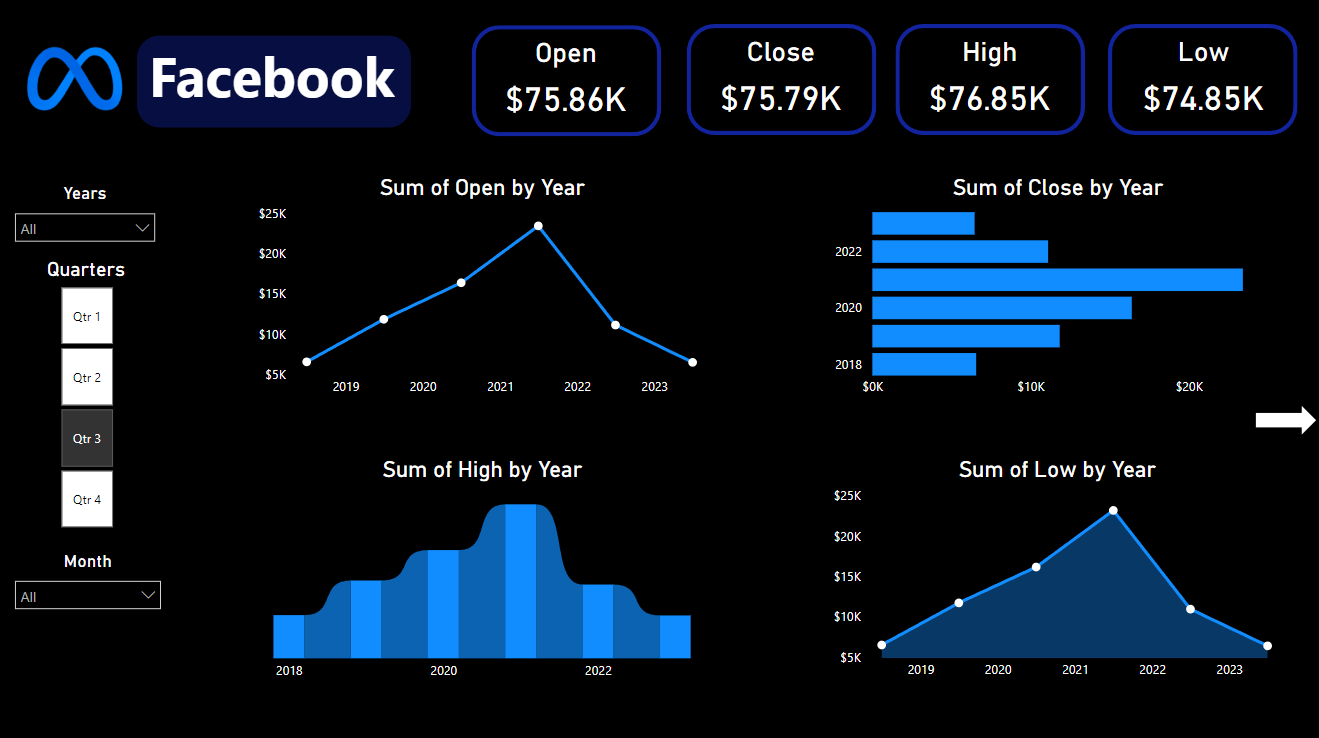
\* Trung bình giá mở cửa và giá đóng cửa theo Công ty



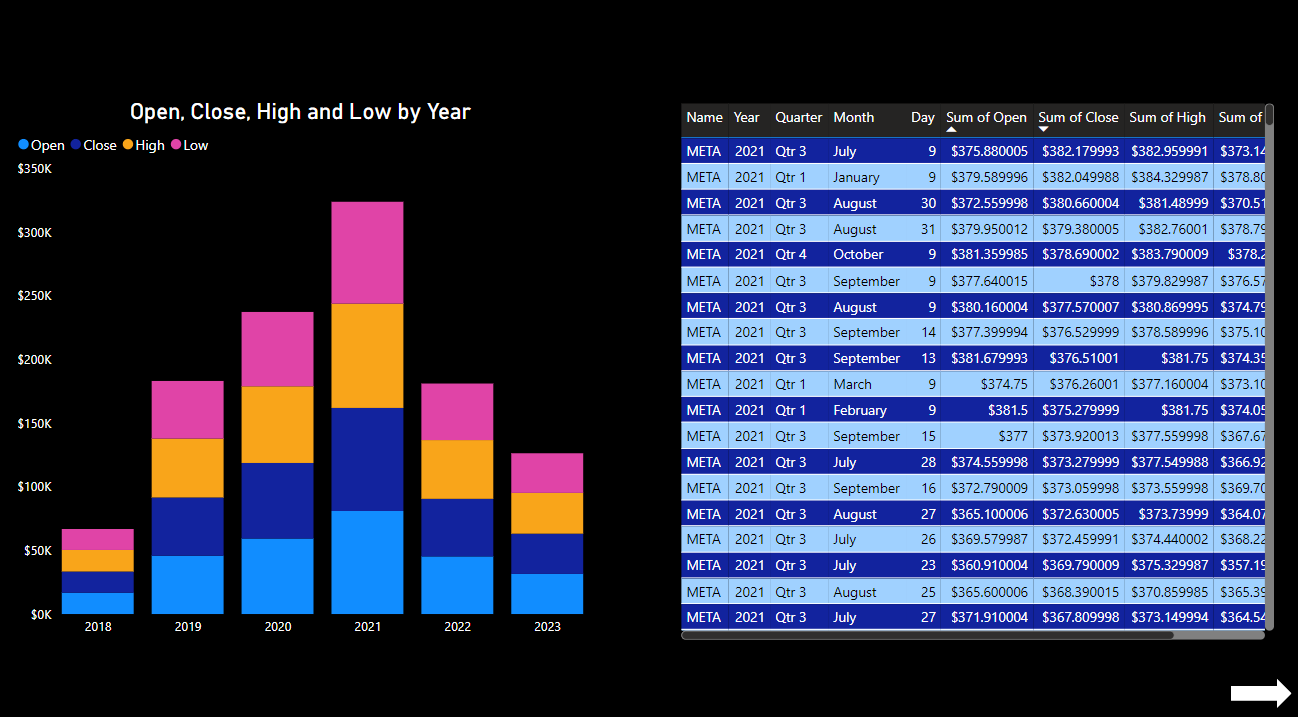
## 2.3.Trực quan trên PowerBI

### 2.3.1 Tập dữ liệu facebook.csv

Tính tổng giá mở cửa, đóng cửa, giá đỉnh và giá thấp theo tháng, theo năm, theo quý

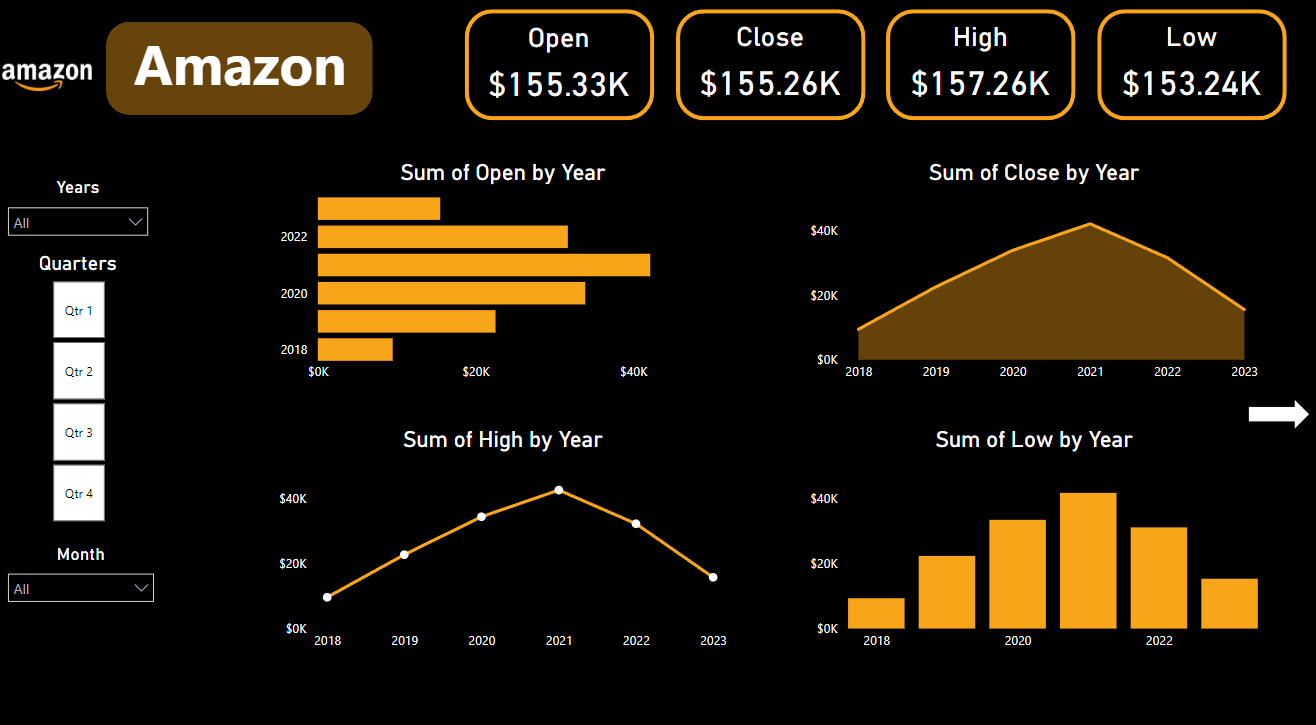


So sánh các tổng giá mở cửa, đóng cửa, giá đỉnh và giá thấp theo năm



### 2.3.2 Tập dữ liệu amazon.csv

Tính tổng giá mở cửa, đóng cửa, giá đỉnh và giá thấp theo tháng, theo năm, theo quý

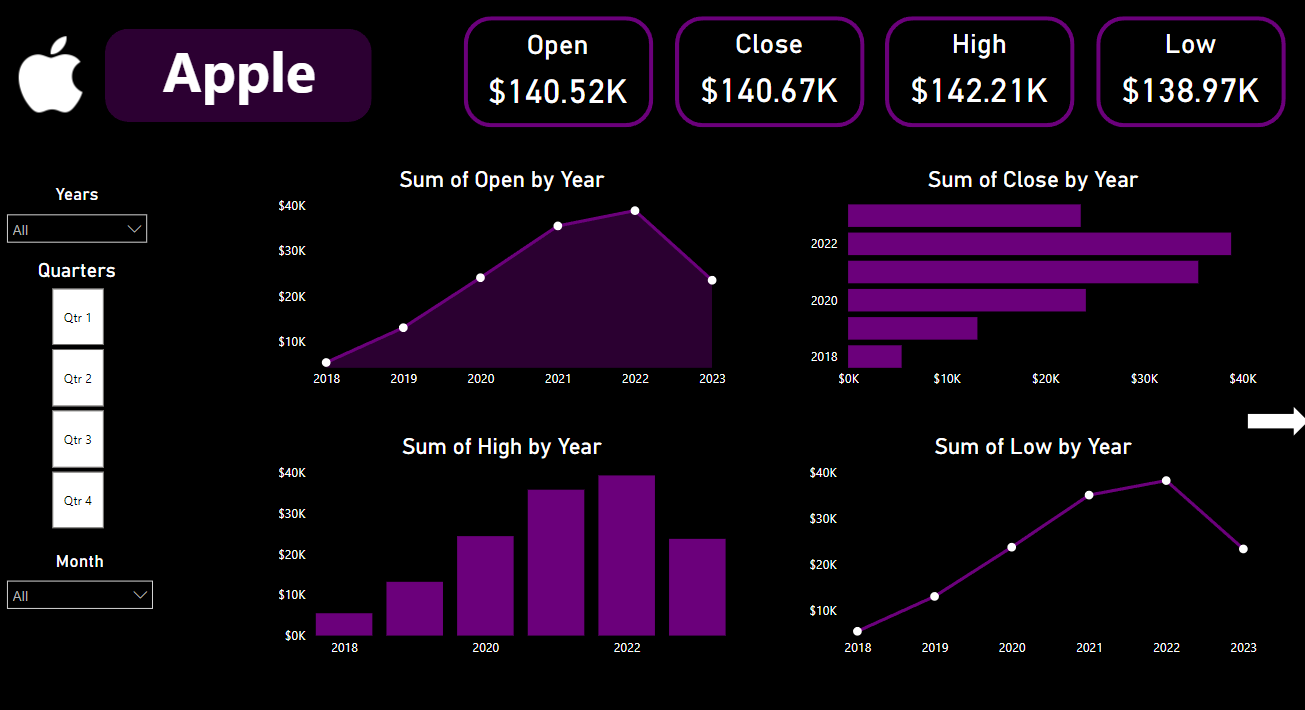


So sánh các tổng giá mở cửa, đóng cửa, giá đỉnh và giá thấp theo năm

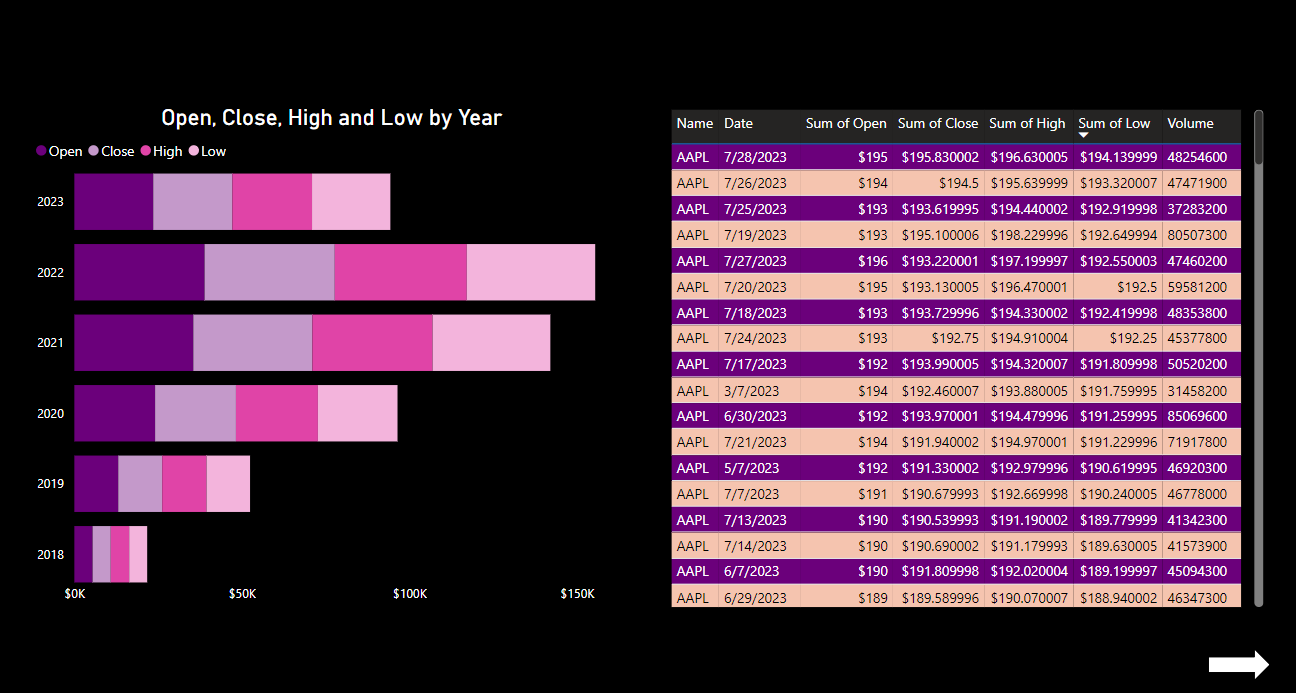


### 2.3.3 Tập dữ liệu Apple.csv

Tính tổng giá mở cửa, đóng cửa, giá đỉnh và giá thấp theo tháng, theo năm, theo quý

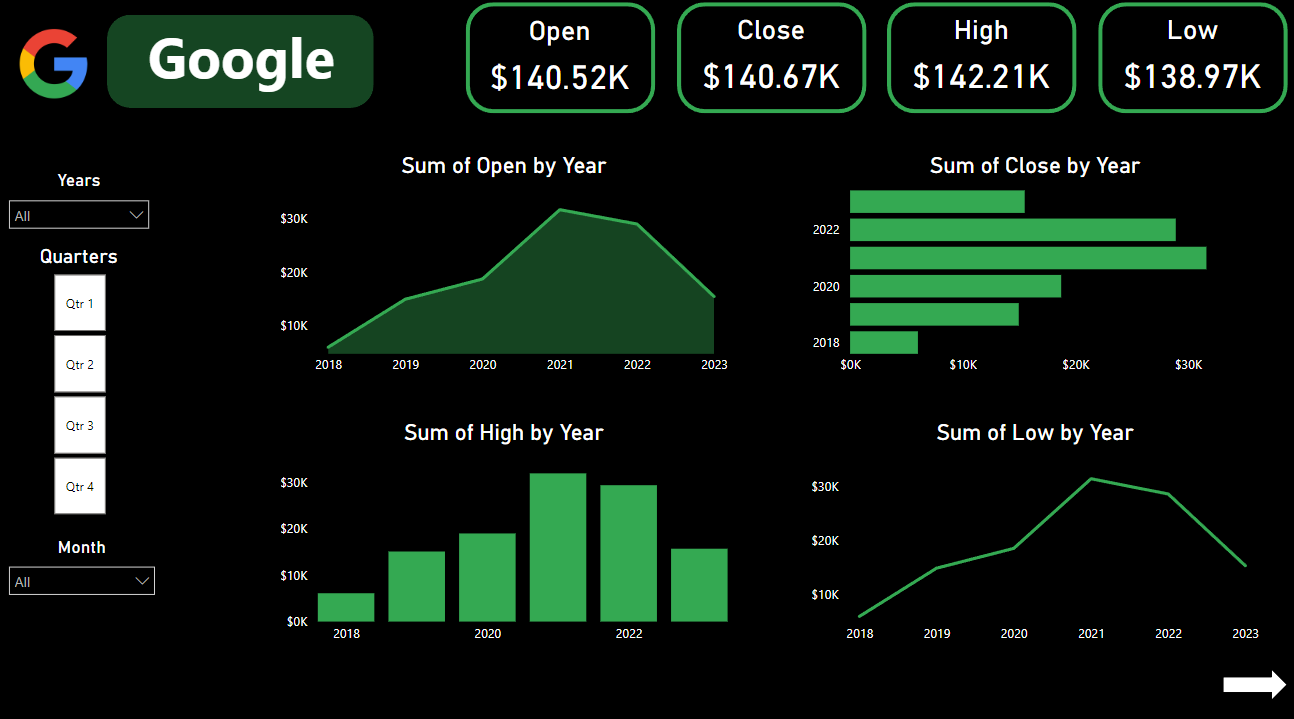


So sánh các tổng giá mở cửa, đóng cửa, giá đỉnh và giá thấp theo năm

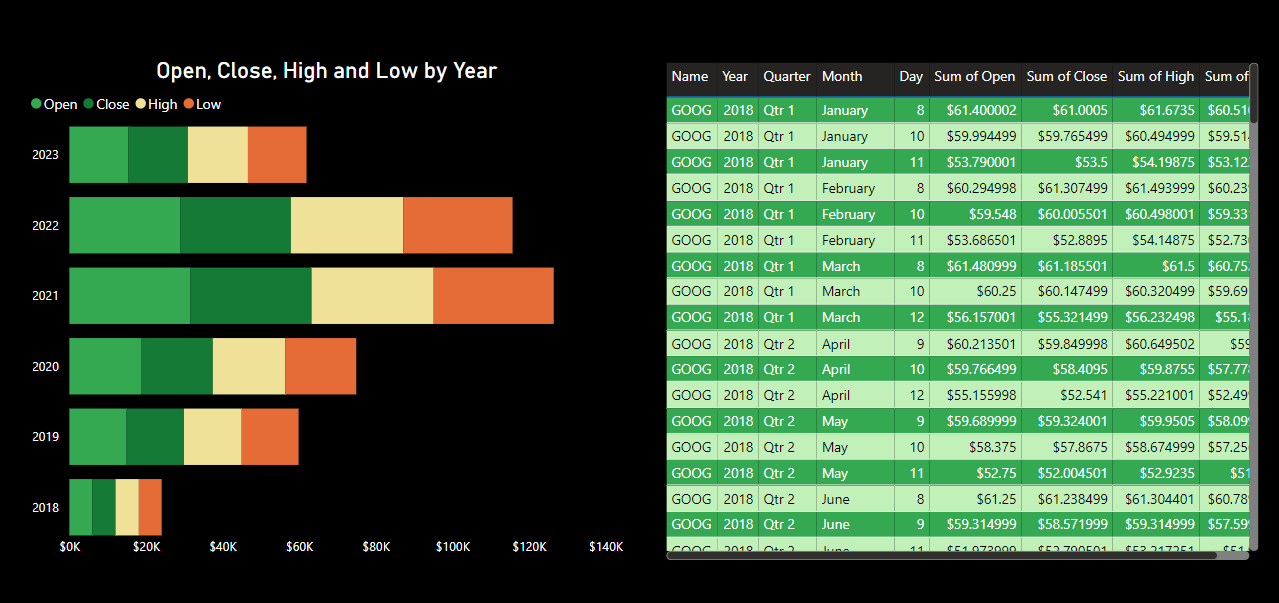


### 2.3.4 Tập dữ liệu Google.csv

Tính tổng giá mở cửa, đóng cửa, giá đỉnh và giá thấp theo tháng, theo năm, theo quý



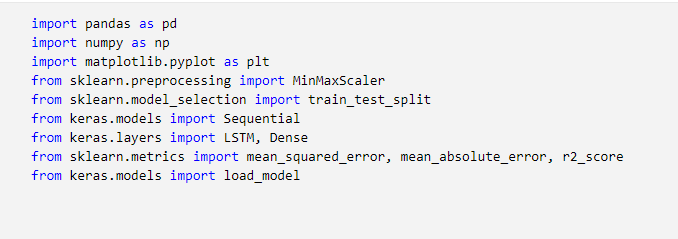
So sánh các tổng giá mở cửa, đóng cửa, giá đỉnh và giá thấp theo năm



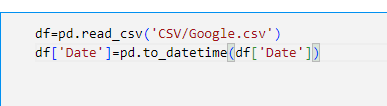
# CHƯƠNG 3: DỰ ĐOÁN DỮ LIỆU

## 3.1.Sử dụng LTSM và RNN dự đoán tập dữ liệu Google

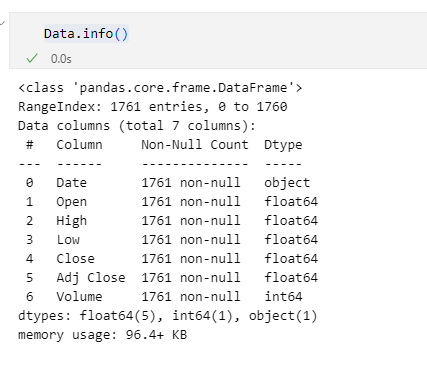
### 3.1.1.Import các thư viện cần thiết



### 3.1.2.Đọc file CSV và Đổi kiểu dữ liệu ngày

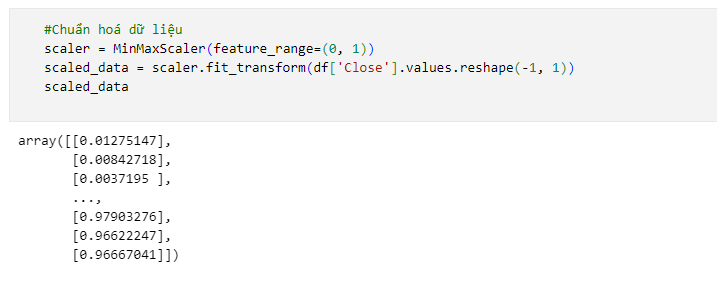


### 3.1.3.Kiểm tra tệp dữ liệu

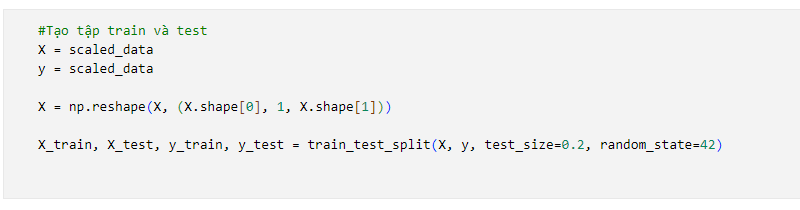


### 3.1.4.Chuẩn bị tập train và set

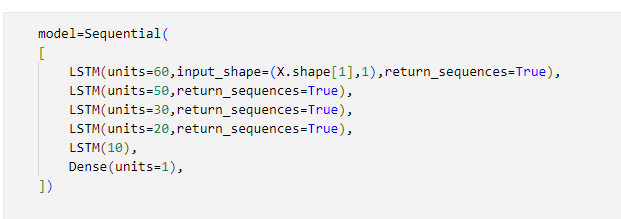
- Chuẩn hoá dữ liệu cột Close giá trị của dữ liệu sẽ được chuẩn hoá nằm trong vùng giá trị từ [0,1]



- Chuản bị tập train và test

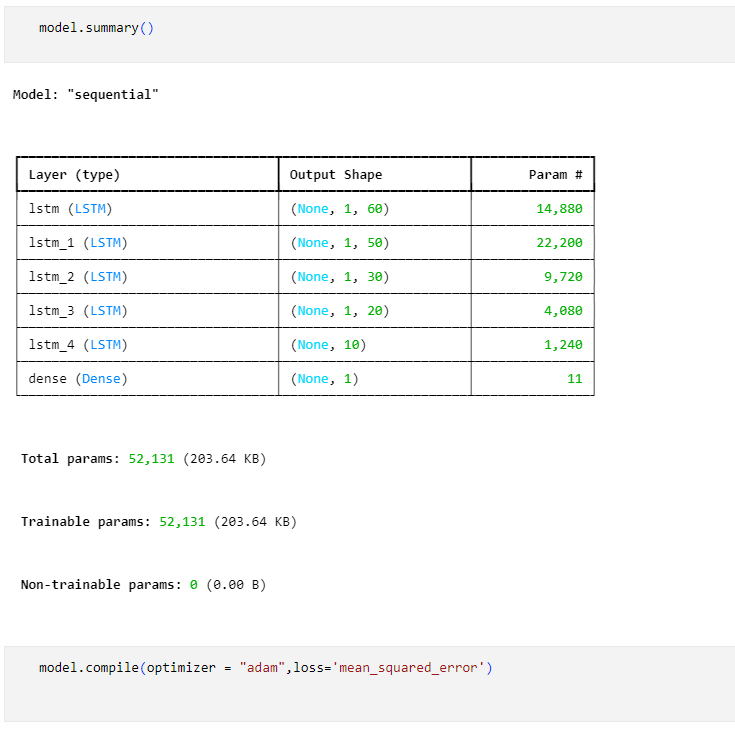


### 3.1.5.Tạo Model

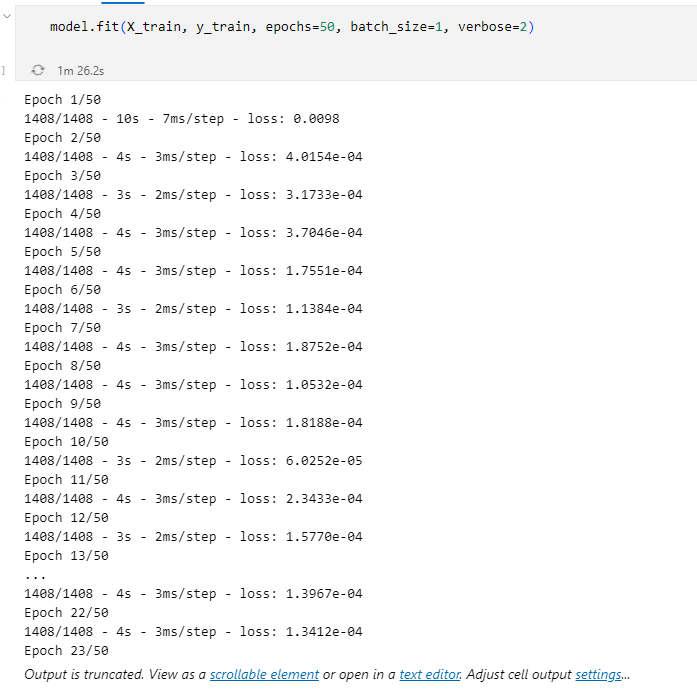


* Lớp đầu tiền
  + LSTM(units=60, input\_shape=(X.shape[1], 1), return\_sequences=True) định nghĩa một lớp LSTM với 60 đơn vị (units).
  + Tham số input\_shape=(X.shape[1], 1) xác định hình dạng đầu vào của lớp LSTM. Trong trường hợp này, kích thước của dữ liệu đầu vào là (X.shape[1], 1), trong đó X.shape[1] đại diện cho độ dài của mỗi cửa sổ dữ liệu và 1 đại diện cho số lượng đặc trưng.
  + return\_sequences=True cho phép lớp LSTM trả về chuỗi kết quả cho lớp LSTM tiếp theo trong mạng nơ-ron.
* Các lớp LSTM tiếp theo:
  + LSTM(units=50, return\_sequences=True) định nghĩa một lớp LSTM với 50 đơn vị (units) và trả về chuỗi kết quả.
  + LSTM(units=30, return\_sequences=True) định nghĩa một lớp LSTM với 30 đơn vị (units) và trả về chuỗi kết quả.
  + LSTM(units=20, return\_sequences=True) định nghĩa một lớp LSTM với 20 đơn vị (units) và trả về chuỗi kết quả.
* Lớp LSTM cuối cùng:
  + LSTM(10) định nghĩa một lớp LSTM với 10 đơn vị (units).
  + Lớp này không có tham số return\_sequences=True, do đó sẽ trả về kết quả duy nhất.
* Lớp Dense:
  + Dense(units=1) định nghĩa một lớp Dense với 1 đơn vị (units).
  + Lớp này sẽ ánh xạ đầu ra từ lớp LSTM cuối cùng sang một giá trị dự đoán duy nhất.

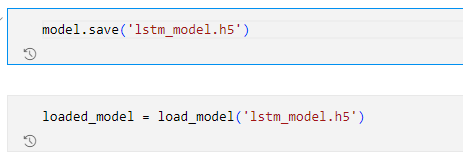
Sử dụng phương pháp tối ưu hóa Adam (Adam optimizer) và hàm mất mát là mean squared error (MSE) để biên dịch (compile) mô hình "Model1".



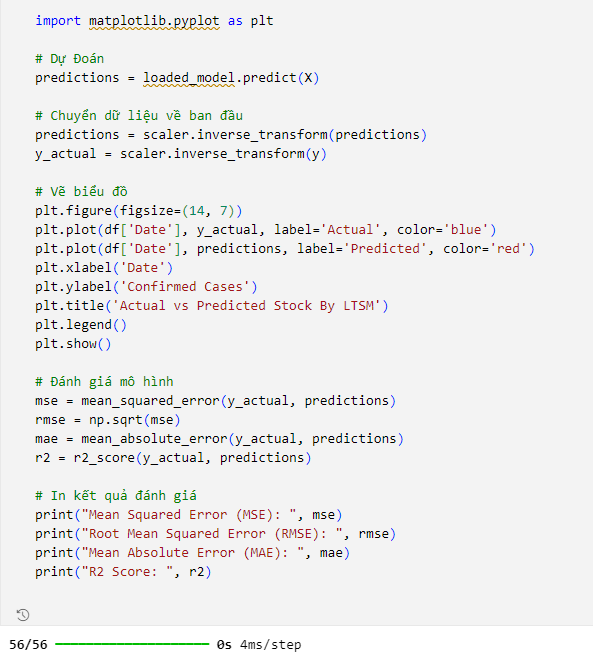
### 3.1.6.Huấn luyện mô hình

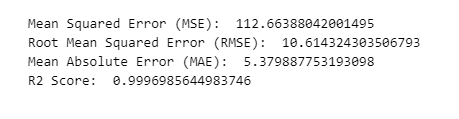


Lưu mô hình lại và load mô hình

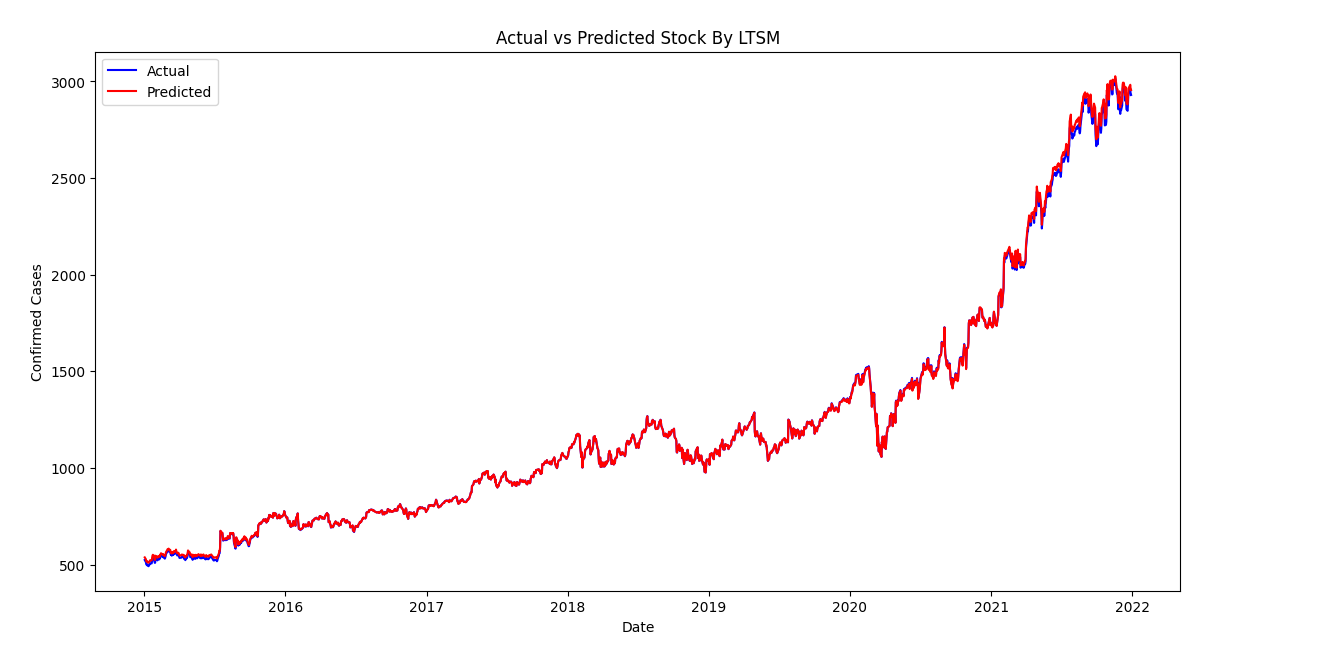


### 3.1.7.Dự đoán và đánh giá các chỉ số sai số



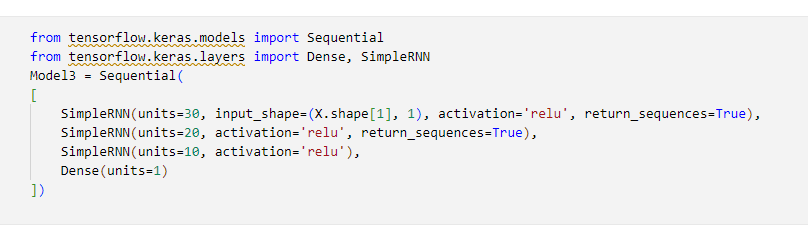


### 3.1.8.Trực quan giá trị dự đoán và thực tế

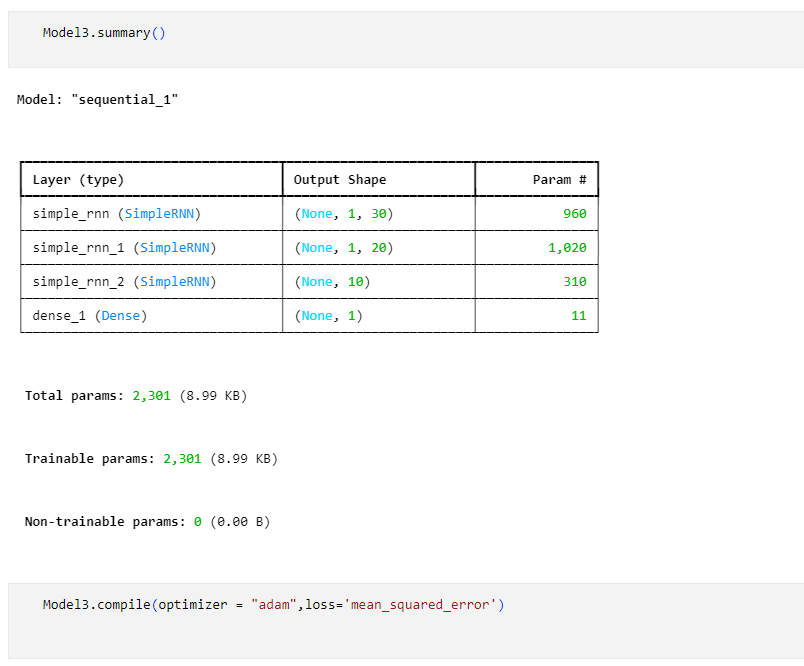


### 3.1.9.Sử dụng RNN dự đoán mô hình

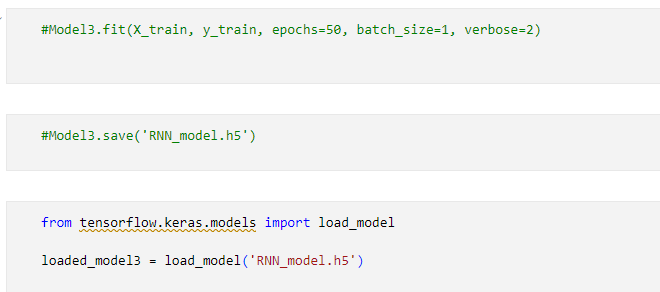
* Import thư viện và tạo model



Sử dụng phương pháp tối ưu hóa Adam (Adam optimizer) và hàm mất mát là mean squared error (MSE) để biên dịch (compile) mô hình "Model3".

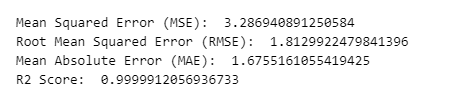


* Thực hiện training sau đó sao lưu và load lại mô hình

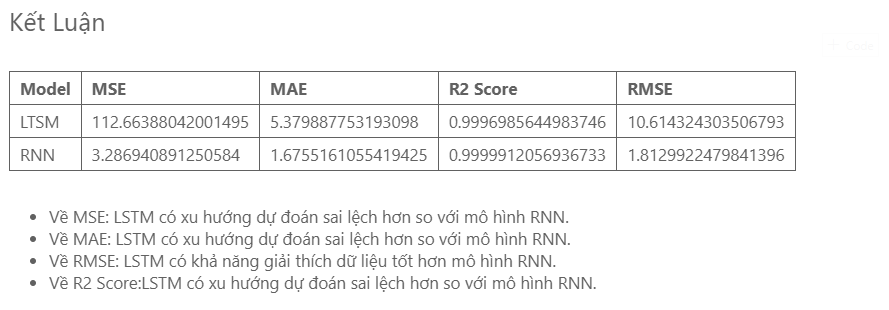


* Thực hiện việc dự đoán và đánh giá các chỉ số mô hình





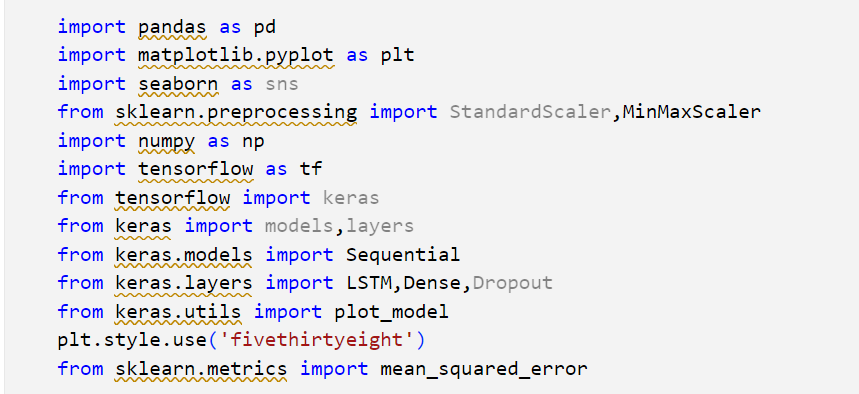
### 3.1.10.So sánh đánh giá 2 mô hình



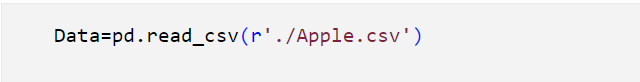
-> Dựa trên ba chỉ số MSE, MAE và RMSE, có thể thấy rằng mô hình RNN có hiệu suất tốt hơn mô hình LSTM. Tuy nhiên, mô hình LSTM có giá trị R2 Score cao hơn, cho thấy khả năng giải thích dữ liệu tốt hơn.

## 3.2.Sử dụng LTSM và Simple Dense Layer dự đoán tập dữ liệu Apple

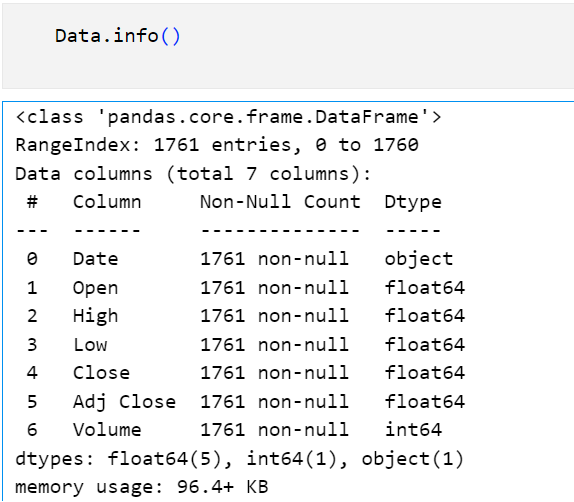
### 3.2.1.Import các thư viện cần thiết



### 3.2.2.Đọc file CSV và In 5 dòng đầu

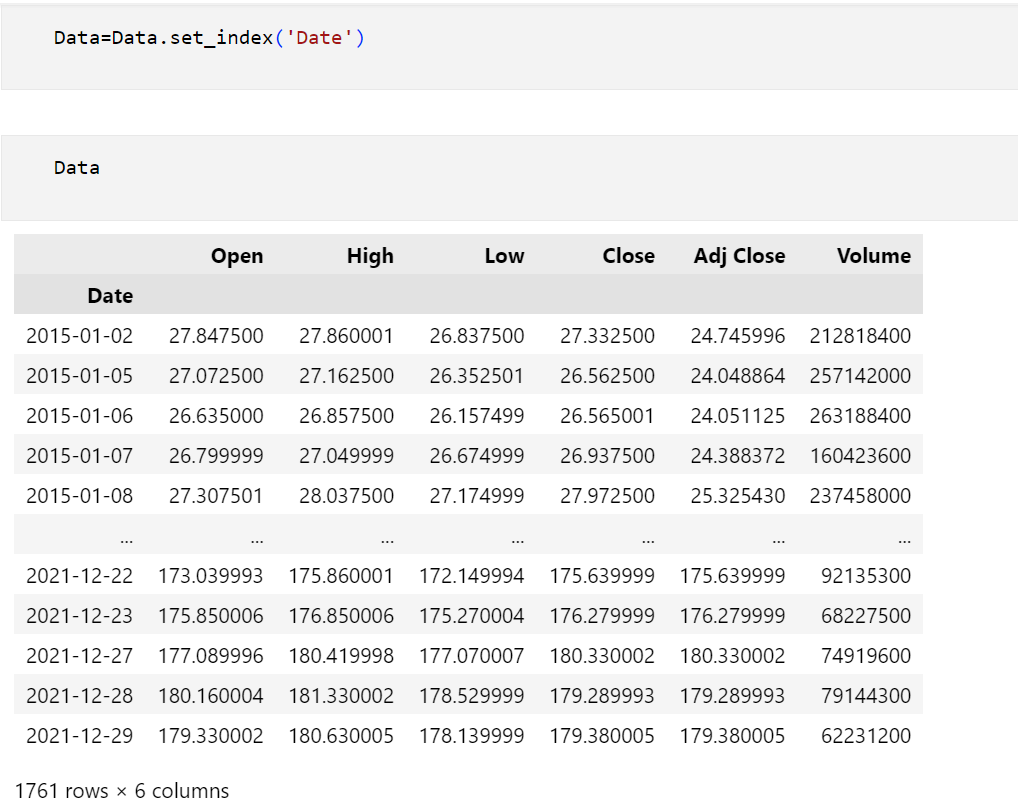


### 3.2.3.Kiểm tra tệp dữ liệu



### 3.2.4.Chuẩn bị tập train và set

* Đặt Date làm chỉ mục

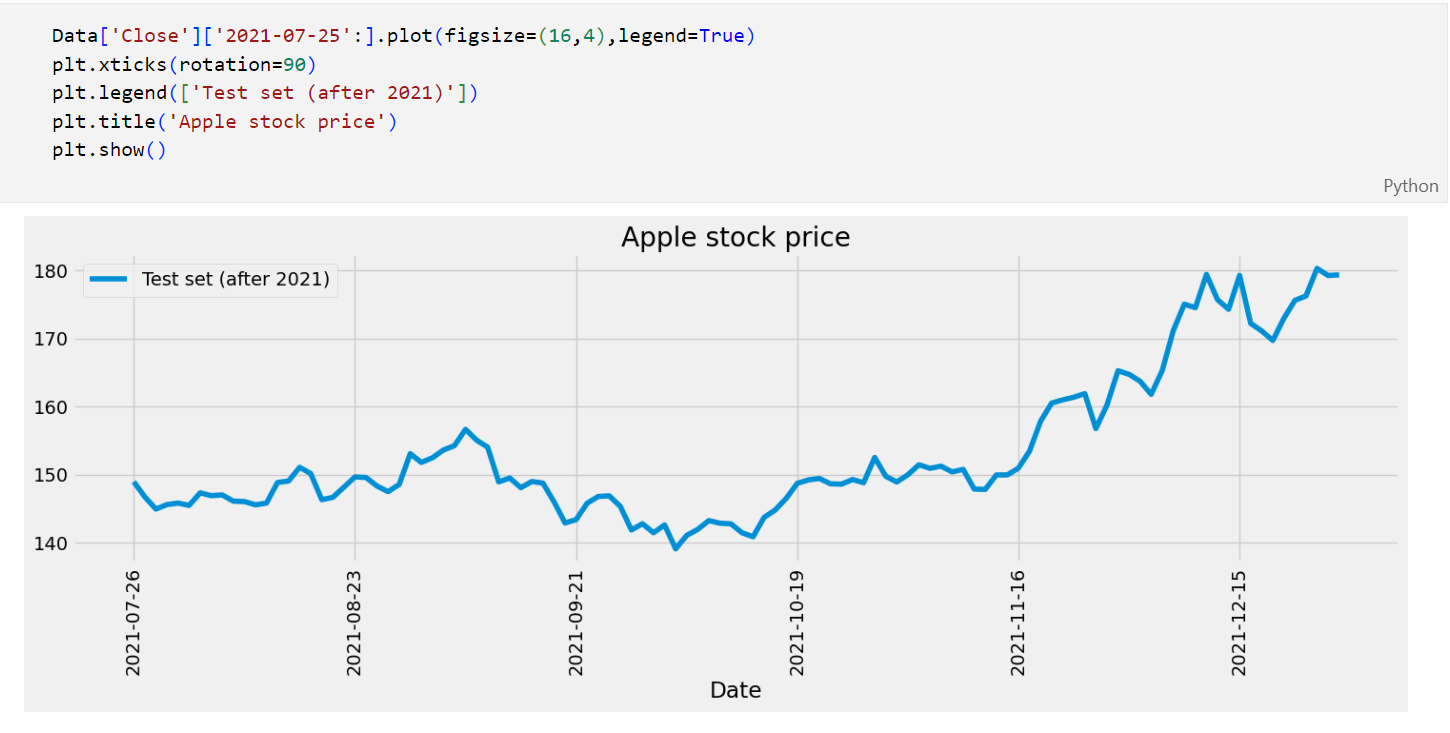


* Tạo tập train và set theo thời gian và tiến hành chuẩn hoá dữ liệu

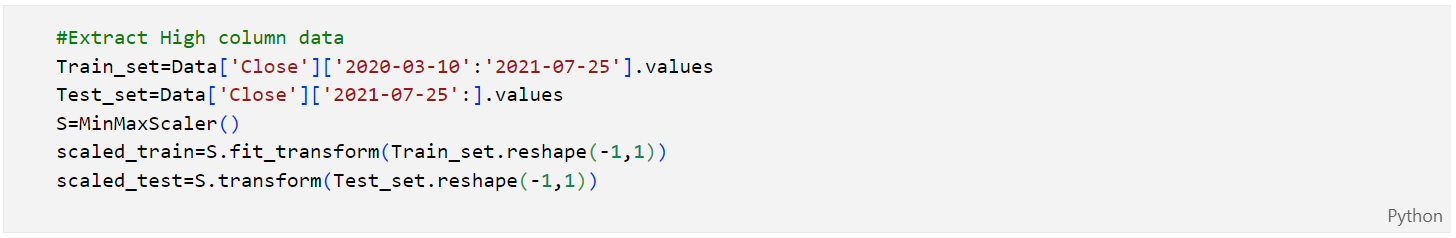


Tạo ra một biểu đồ giá cổ phiếu của Apple trong khoảng thời gian từ năm 2020 đến ngày 25 tháng 7 năm 2021, với chú thích chỉ ra rằng đây là tập dữ liệu huấn luyện.

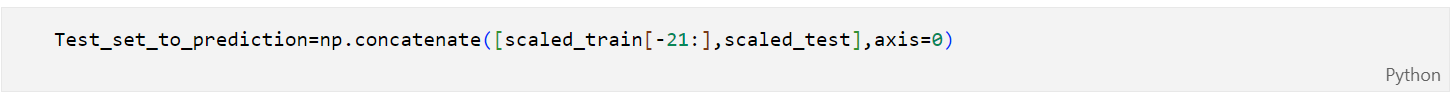
* Tạo tập test và set theo thời gian và tiến hành chuẩn hoá dữ liệu



Tạo ra một biểu đồ giá cổ phiếu của Apple từ ngày 25 tháng 7 năm 2021 đến hiện tại, với chú thích chỉ ra rằng đây là tập test sau năm 2021.



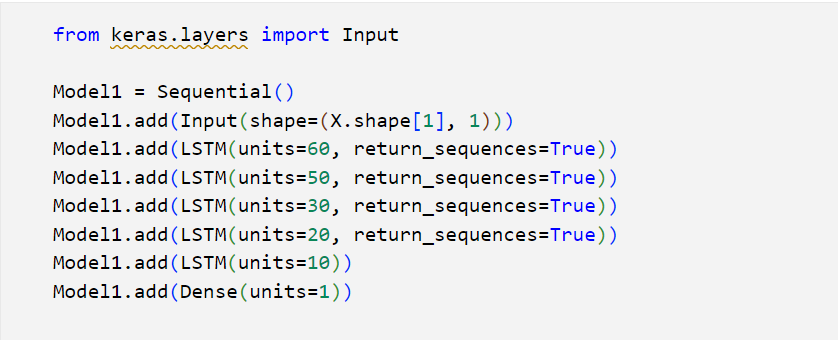
Trích xuất dữ liệu cột "Close" từ DataFrame "Data" trong khoảng thời gian từ ngày 10 tháng 3 năm 2020 đến ngày 25 tháng 7 năm 2021. Dữ liệu này được lưu trữ trong biến "Train\_set". Và trích xuất dữ liệu cột "Close" từ DataFrame "Data" từ ngày 25 tháng 7 năm 2021 đến hết dữ liệu. Dữ liệu này được lưu trữ trong biến "Test\_set".



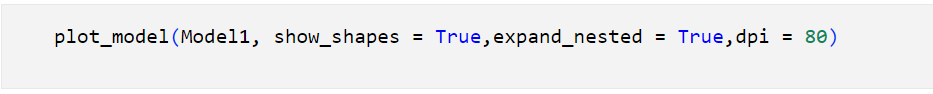
Chuẩn bị dữ liệu để dự đoán trên tập test bằng mô hình Recurrent Neural Network (RNN). Trong đó, RNN hoạt động trên dữ liệu chuỗi, và để dự đoán trên tập test, mô hình cần nhìn thấy dữ liệu từ tập train. Cụ thể, 20 dữ liệu cuối cùng từ tập train sẽ được sử dụng để dự đoán điểm dữ liệu đầu tiên trong tập test.

Dòng mã scaled\_train[-21:] lấy 20 dữ liệu cuối cùng từ tập train, và scaled\_test là tập test. Hàm np.concatenate được sử dụng để ghép hai tập dữ liệu này lại với nhau theo trục 0 (axis=0), tức là ghép theo chiều dọc. Kết quả chúng ta nhận được là Test\_set\_to\_prediction, đó là tập dữ liệu đã được chuẩn bị để dự đoán trên tập test.

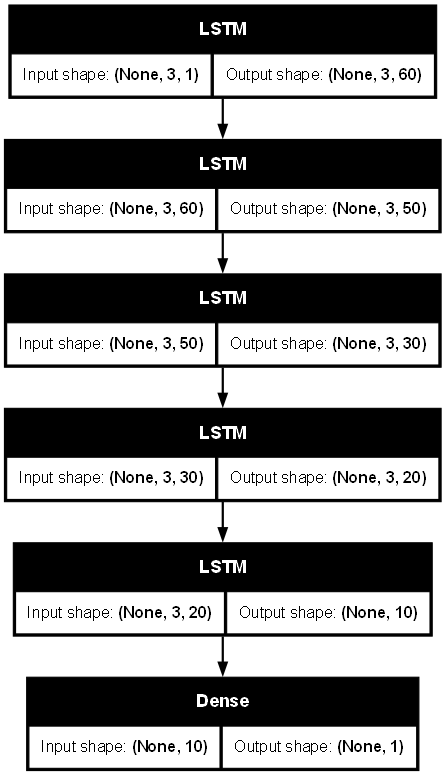
### 3.2.5.Tạo Model



Tạo ra một mô hình RNN đã được xây dựng và sẵn sàng để được huấn luyện và sử dụng cho các tác vụ dự đoán dữ liệu chuỗi.

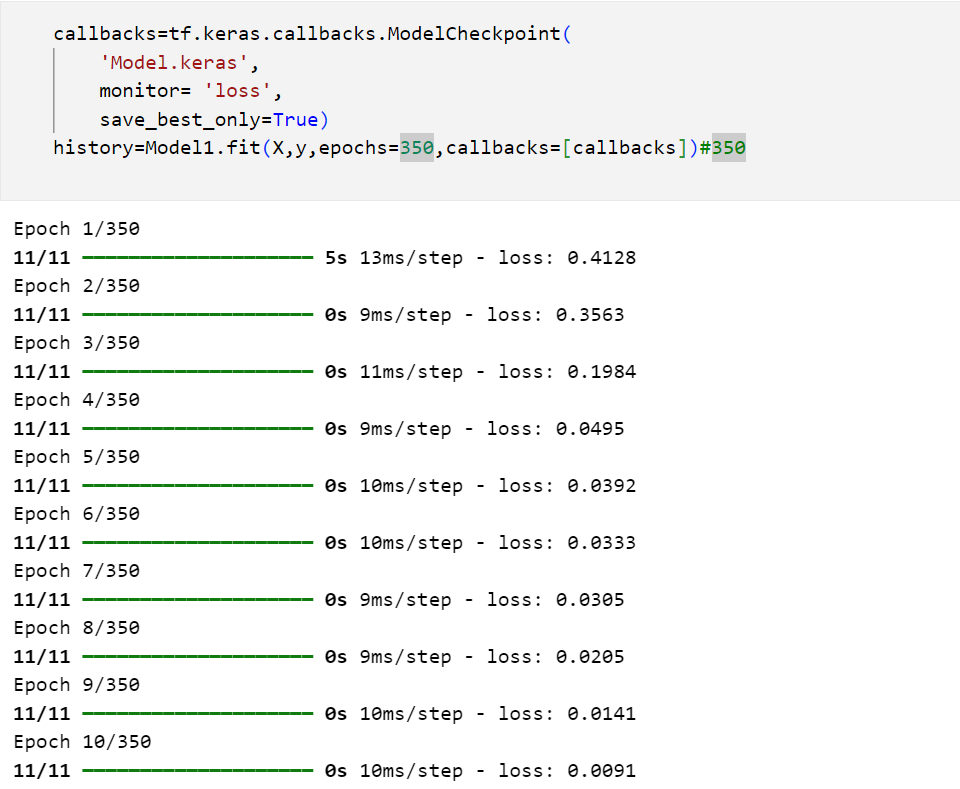


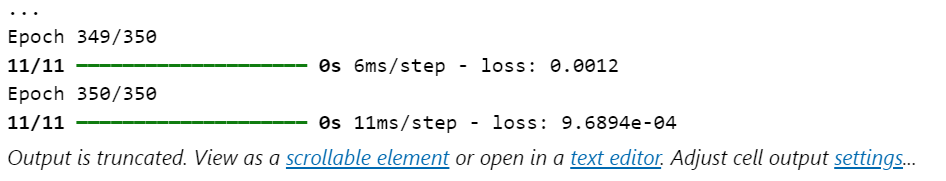
Vẽ biểu đồ kiến trúc của mô hình Model1



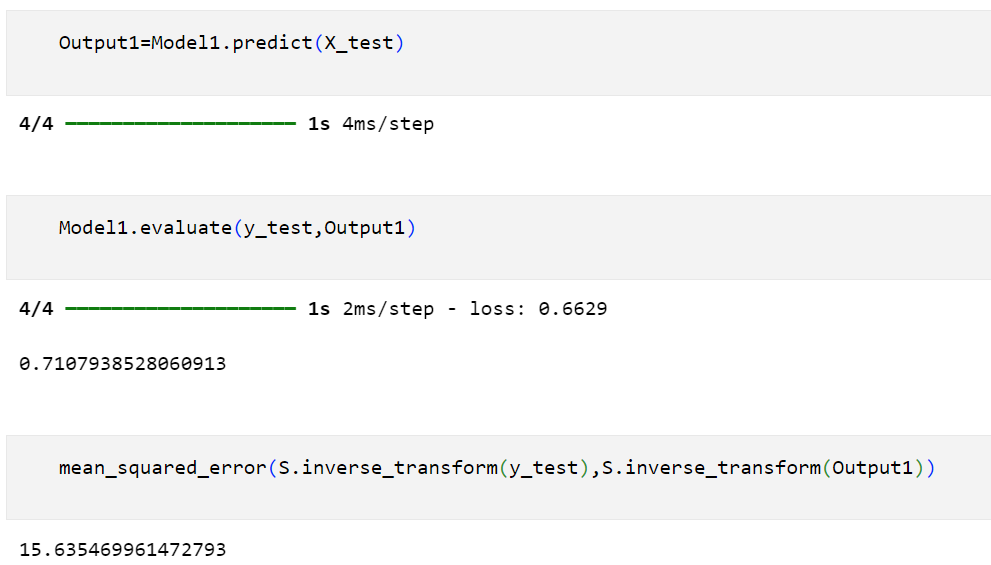
### 3.2.6.Huấn luyện mô hình

Huấn luyện mô hình Model1 sử dụng dữ liệu X và y trong epochs vòng lặp.





### 3.2.7.Dự đoán



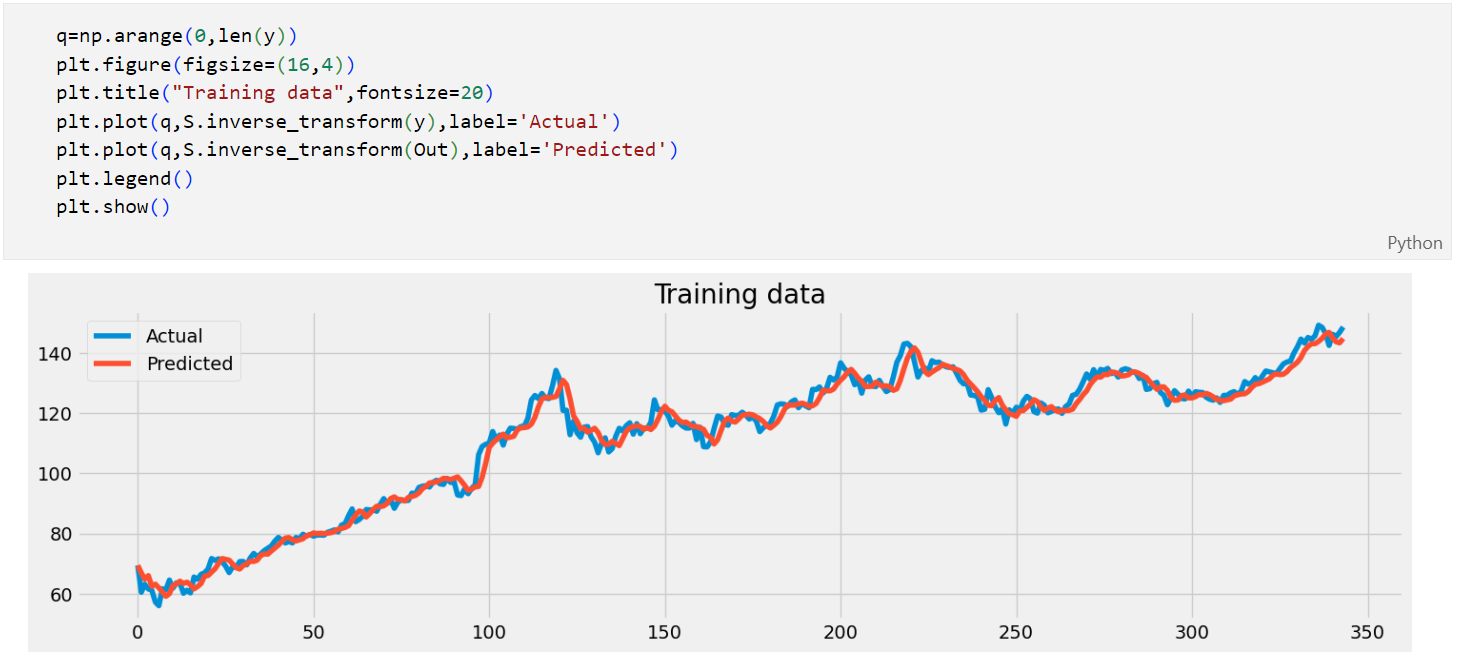
* Sử dụng mô hình "Model1" để dự đoán đầu ra cho dữ liệu kiểm thử "X\_test". Kết quả dự đoán được lưu vào biến "Output1".
* Phương thức evaluate() trong mô hình Keras được sử dụng để đánh giá hiệu suất của mô hình.
* Để tính toán mean squared error (MSE) giữa dự đoán của mô hình và nhãn thực tế trên dữ liệu kiểm thử.

### 3.2.8. Trực quan giá trị dự đoán và thực tế

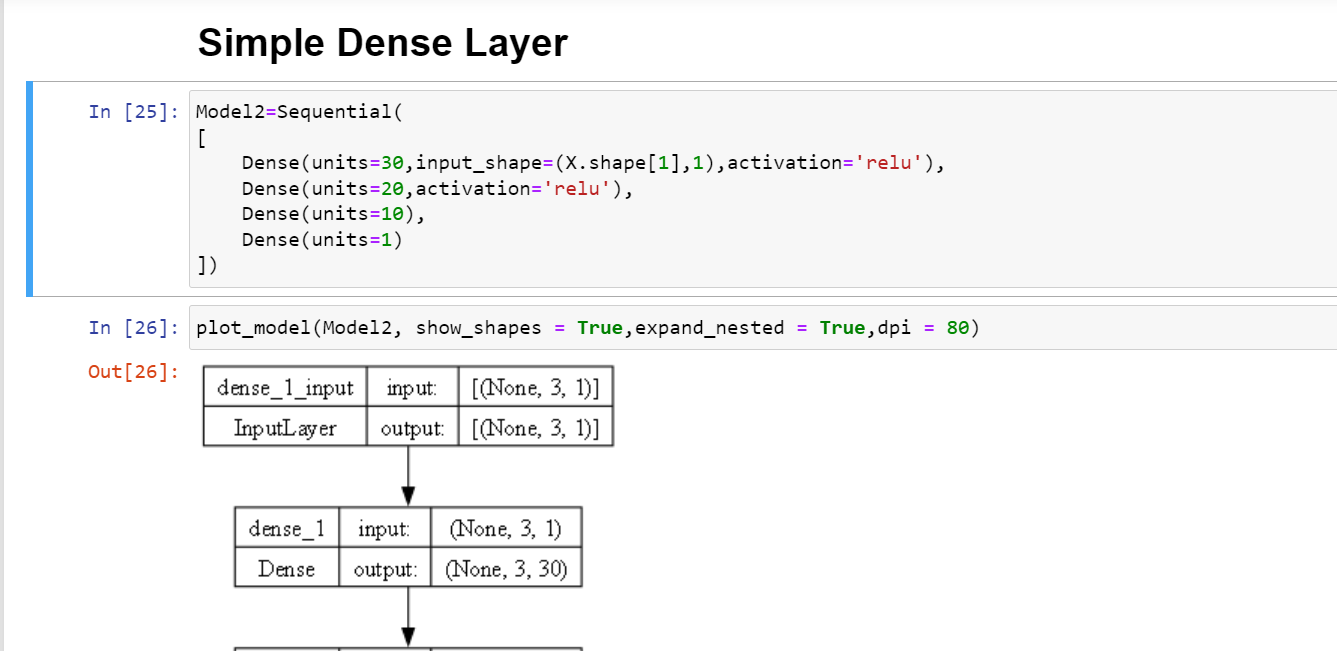
Đồ thị so sánh giữa dữ liệu thực tế và dự đoán của mô hình LSTM trên tập dữ liệu kiểm tra.



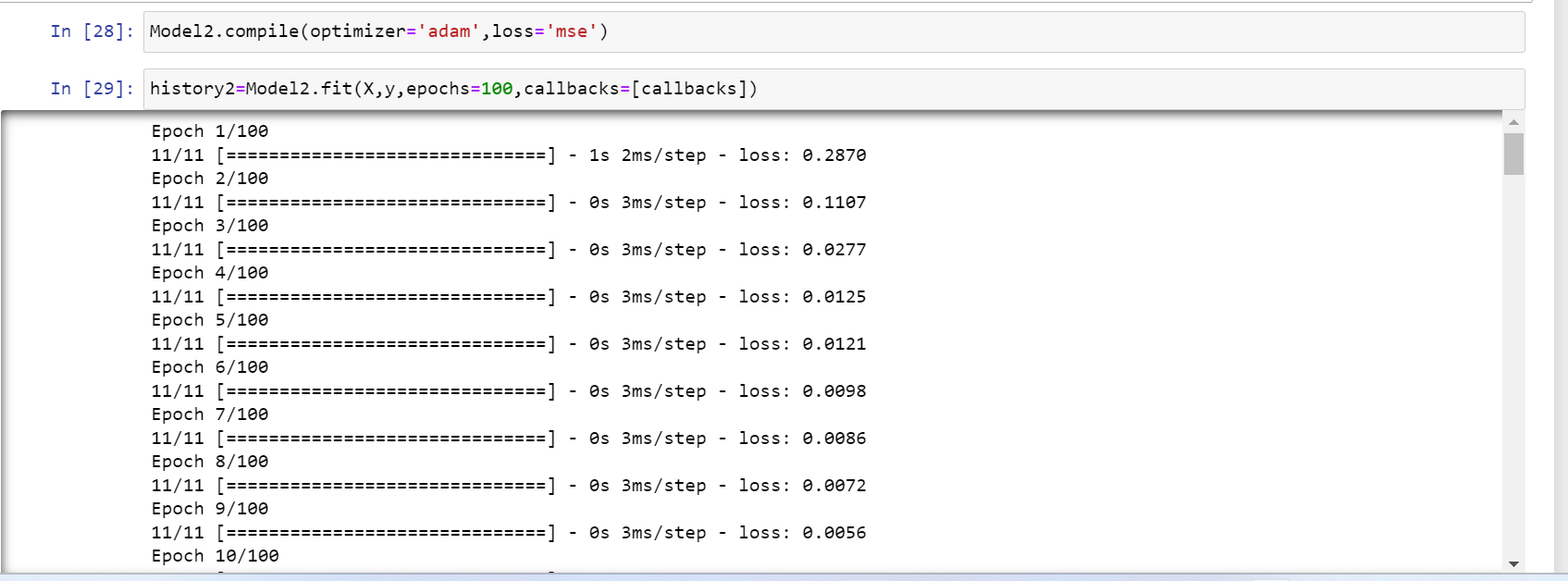
Đồ thị so sánh giữa dữ liệu thực tế và dự đoán của mô hình trên tập dữ liệu huấn luyện.



### 3.2.9. Sử dụng mô hình Simple Dense Layer để so sánh hiệu suất đánh giá của LSTM

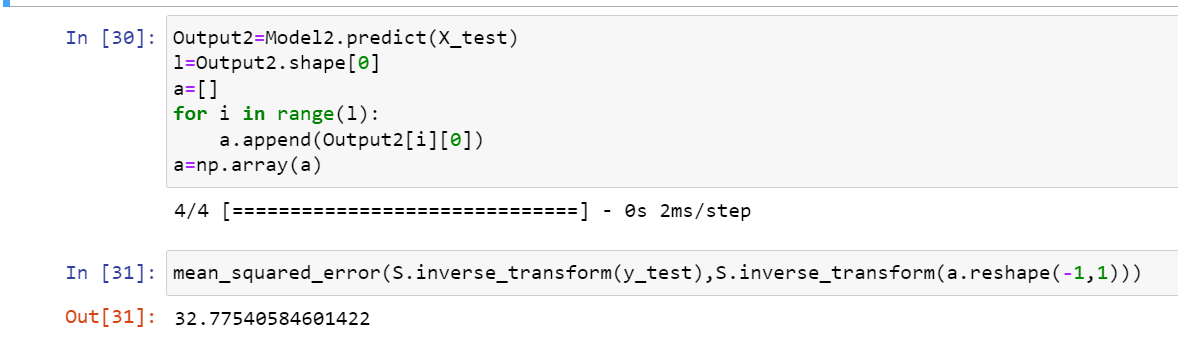


Tạo ra một mô hình neural network gồm 4 tầng Dense, và sau đó vẽ ra biểu đồ mô tả cấu trúc của mô hình, giúp người dùng dễ dàng hình dung và hiểu rõ về cấu trúc của mô hình này.

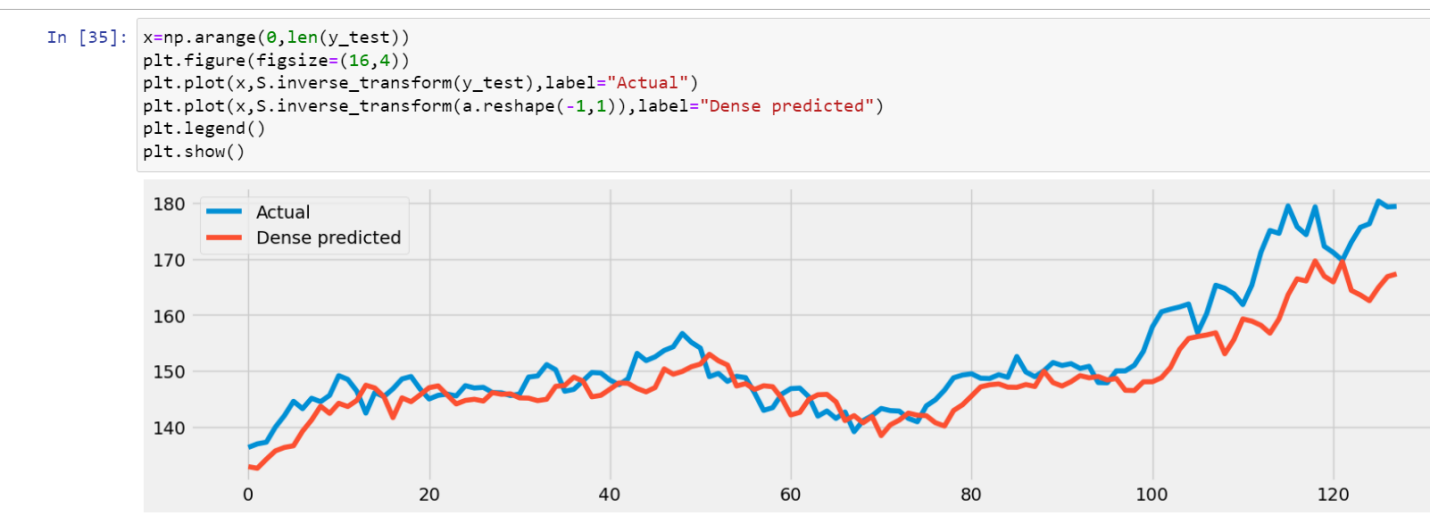


Sau khi thực hiện các bước trên, mô hình Model2 sẽ được huấn luyện sử dụng thuật toán tối ưu hóa Adam và hàm mất mát MSE. Quá trình huấn luyện sẽ diễn ra trong 100 epochs, trong đó các callbacks sẽ được gọi để theo dõi và thực hiện các tác vụ tương ứng. Lịch sử huấn luyện của mô hình sẽ được lưu lại trong biến history2.

Cuối cùng, có thể sử dụng các thông tin trong history2 để đánh giá và theo dõi quá trình huấn luyện của mô hình.

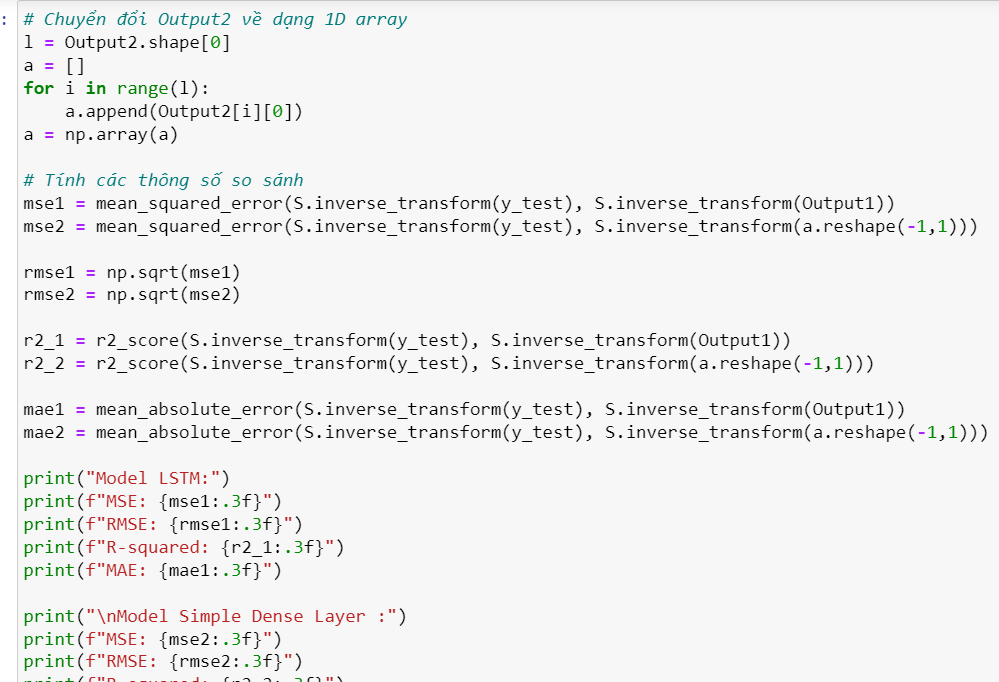


Sử dụng mô hình Model2 để dự đoán trên tập dữ liệu kiểm tra X\_test. Lấy kết quả dự đoán và lưu vào biến Output2. Tách riêng các giá trị dự đoán từ Output2 và lưu vào một mảng NumPy a.



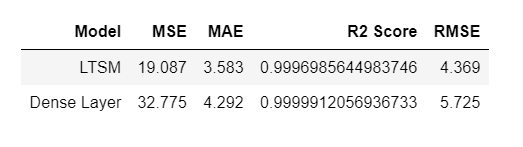
Vẽ một đồ thị so sánh giữa giá trị thực tế (y\_test) và giá trị dự đoán (a) trên tập dữ liệu kiểm tra. Đồ thị sẽ hiển thị đường thực tế và đường dự đoán, cho phép người dùng so sánh và đánh giá kết quả của mô hình.

### 3.2.10. So sánh 2 mô hình áp dụng và đưa ra kết luận



Tính các thông số so sánh giữa mô hình "LSTM" và mô hình "Dense"

**Kết quả:**



Khi so sánh 2 mô hình này, ta có thể thấy rằng:

MSE và RMSE của Model LSTM thấp hơn so với Model Simple Dense Layer, điều này chỉ ra rằng Model LSTM có độ lỗi trung bình thấp hơn, có nghĩa là dự đoán của nó gần với giá trị thực tế hơn.

Giá trị R-squared của Model LSTM là 0.827, cao hơn so với 0.704 của Model Simple Dense Layer. R-squared là một chỉ số đánh giá mức độ phù hợp của mô hình, giá trị càng cao (gần 1) thì mô hình càng tốt. Vì vậy, Model LSTM có khả năng giải thích và dự đoán tốt hơn so với Model Simple Dense Layer.

MAE của Model LSTM là 3.583, thấp hơn so với 4.292 của Model Simple Dense Layer. MAE đo lường độ lệch trung bình tuyệt đối giữa giá trị dự đoán và giá trị thực tế, do đó, Model LSTM có độ chính xác cao hơn.

**Kết luận:**

Dựa trên các chỉ số đánh giá, Model LSTM có hiệu suất tốt hơn so với Model Simple Dense Layer.

Mô hình LSTM thể hiện khả năng dự đoán và giải thích cao hơn, với độ lỗi trung bình thấp hơn và độ chính xác cao hơn.

Do đó, có thể kết luận rằng Model LSTM là mô hình tốt hơn để giải quyết bài toán này.

## CHƯƠNG 4: TỔNG KẾT

* Trong nghiên cứu "Phân tích giá cổ phiếu áp dụng LSTM để dự đoán giá cổ phiếu," chúng ta đã tiến hành một cuộc nghiên cứu toàn diện nhằm ứng dụng mô hình LSTM (Long Short-Term Memory) vào việc dự đoán giá cổ phiếu dựa trên dữ liệu lịch sử. Sử dụng tập dữ liệu lịch sử giá cổ phiếu của 4 công ty phổ biến từ Kaggle, chúng ta đã thực hiện các bước tiền xử lý dữ liệu cần thiết, bao gồm làm sạch và chuẩn hóa dữ liệu. Mô hình LSTM đã được huấn luyện và tinh chỉnh để tối ưu hóa khả năng dự đoán.
* Kết quả cho thấy mô hình LSTM có khả năng ghi nhớ và học từ dữ liệu lịch sử, giúp dự đoán giá cổ phiếu với độ chính xác tương đối cao, mặc dù vẫn còn một số sai số do biến động khó lường của thị trường. Các chỉ số đánh giá như MAE, RMSE và R² đã khẳng định hiệu suất của mô hình. Nghiên cứu này không chỉ minh chứng cho tiềm năng ứng dụng của LSTM trong lĩnh vực tài chính mà còn cung cấp nền tảng cho các nghiên cứu và phát triển tiếp theo.
* Hơn nữa, nghiên cứu đã giúp chúng ta hiểu sâu hơn về cách áp dụng LSTM vào dữ liệu chuỗi thời gian, từ đó mở ra nhiều hướng phát triển mới như tích hợp thêm các yếu tố kinh tế vĩ mô và dữ liệu tin tức để cải thiện chất lượng dự đoán. Tóm lại, việc áp dụng LSTM trong dự đoán giá cổ phiếu đã cho thấy tiềm năng to lớn trong việc hỗ trợ các nhà đầu tư và nhà phân tích tài chính ra quyết định chính xác hơn, đồng thời góp phần vào sự phát triển của các công nghệ dự đoán hiện đại trong lĩnh vực tài chính.

## CHƯƠNG 5:TÀI LIỆU THAM KHẢO

[1] *LTSM là gì*?, Hai`s blog,20/10/2017, <https://dominhhai.github.io/vi/2017/10/what-is-lstm/>

[2] *Machine Learning to Predict Stock Prices* , Roshan Adusumilli, Dec 26, 2019, <https://towardsdatascience.com/predicting-stock-prices-using-a-keras-lstm-model-4225457f0233>

[3] *Stock Market Predictions with LSTM in Python,*DataCamp,2020 Januarary, <https://www.datacamp.com/tutorial/lstm-python-stock-market>

[4] *LSTM for Apple stock prediction*,kaggle, <https://www.kaggle.com/code/anikbhowmick/lstm-for-apple-stock-prediction>