Note on yfinance Python library

The yfinance is a Python library that provides a user-friendly interface for downloading historical market data from Yahoo Finance. It allows you to get historical stock prices, dividends, and other financial data for stocks, Exchange-Traded Funds (ETFs), and other securities.

Code:

```
import yfinance as yf
import pandas as pd
# Download historical data for a stock
apple = yf.Ticker("AAPL")

import json
with open('apple.json') as json_file:
    apple_info = json.load(json_file)
    # Print the type of data variable
    #print("Type:", type(apple_info))
apple_info
```

Output:

```
Out[50]: {'zip': '95014',
          'sector': 'Technology',
          'fullTimeEmployees': 100000,
          'longBusinessSummary': 'Apple Inc. designs, man
         ufactures, and markets smartphones, personal com
         puters, tablets, wearables, and accessories worl
         dwide. It also sells various related services. I
         n addition, the company offers iPhone, a line of
         smartphones; Mac, a line of personal computers;
         iPad, a line of multi-purpose tablets; AirPods M
         ax, an over-ear wireless headphone; and wearable
         s, home, and accessories comprising AirPods, App
         le TV, Apple Watch, Beats products, HomePod, and
         iPod touch. Further, it provides AppleCare suppo
         rt services; cloud services store services; and
         operates various platforms, including the App St
         ore that allow customers to discover and downloa
         d applications and digital content, such as book
         s, music, video, games, and podcasts. Additional
```

Code:

apple_info[‘country’]

Output:

'United States’

Code:

```
apple_share_price_data = apple.history(period="3mo")
# Display the downloaded data
apple_share_price_data.head()
```

Output:

Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
2023-07-14 00:00:00-04:00	189.973460	190.922176	189.374278	190.432846	41573900	0.0	0.0
2023-07-17 00:00:00-04:00	191.641201	194.057950	191.551326	193.728394	50520200	0.0	0.0
2023-07-18 00:00:00-04:00	193.089258	194.067932	192.160504	193.468735	48353800	0.0	0.0
2023-07-19 00:00:00-04:00	192.839596	197.962667	192.390191	194.836899	80507300	0.0	0.0
2023-07-20 00:00:00-04:00	194.826901	196.205045	192.240397	192.869553	59581200	0.0	0.0

First, I imported the yfinance library using the alias yf.
Then, I created a Ticker object for the Apple stock (“AAPL”).
I used the history method of the Ticker object to download the historical data for the stock. The period parameter of the history method specifies the time period for which we want to download the data. In this example, I set it to 3mo to download the amount of available historical data for the past 3 months.

period=”1d”: Download 1 day of historical data.
period=”5d”: Download 5 days of historical data.
period=”1mo”: Download 1 month of historical data.
period=”3mo”: Download 3 months of historical data.
period=”6mo”: Download 6 months of historical data.
period=”1y”: Download 1 year of historical data.
period=”2y”: Download 2 years of historical data.
period=”5y”: Download 5 years of historical data.
period=”10y”: Download 10 years of historical data.
period=”ytd”: Download historical data since the beginning of the current year.
period=”max”: Download all available historical data.

Finally, I printed the downloaded data using the head function. This displayed a Pandas DataFrame containing the historical stock prices and other financial data for Apple.