**[api/Stocks]**

The *StocksController* has 10 overloaded methods that will return a List in XML form of the data requested by your parameters.

The format for calling this API, just like for every other method is formatted as:

**../api/Stocks? Parameter1=value1&Parameter2=value2&…**

(continued in the same format for as many parameters as required.)

***OVERLOADS*** *–* Return List format: { DateTime date, decimal price }

(string symbol, string start, string end)

//returns closing stock price daily for the dates *start* to *end*

(string symbol, string start)

//returns closing stock price daily for the dates *start* to Today

(string symbol, int years)

//returns closing stock price daily for the past x years indicated by int years

(string symbol)

//returns closing stock price daily for the past year.

(string symbol, string start, string end, string interval)

//returns closing stock price at the stated interval, for the dates *start* to *end*

***DETAILED OVERLOADS*** *–* Return List format:

{DateTime date, decimal Open, decimal High, decimal Low, decimal Close, decimal AdjClose}

(string symbol, string start, string end, string interval, bool detailed)

//returns all stock data listed above at the given interval, for the dates *start* to *end*

(string symbol, string start, string interval, bool detailed)

//returns all stock data listed above at the given interval, for the dates *start* to *Today*

(string symbol, string interval, bool detailed)

//returns all stock data listed above at the given interval, for the past year

(string symbol, int years, bool detailed)

//returns all stock data listed above daily for the past x year indicated by int years

(string symbol, bool detailed)

//returns all stock data listed above daily for the past year

***ALL PARAMETERS –*** *required formats:*

string symbol: must be an actual stock symbol

string start: Keep in DateTime format = YYYY-MM-DD

string end: Keep in DateTime format = YYYY-MM-DD

string interval: must be one character { d, w, m } which stand for day, week, and month respectively.

Int years: just a normal int. However, stocks can only go back as far as they exist.

Bool detailed: Can be true or false. All it does is indicate that you are using a detailed overload.

**EXAMPLE CALL: ../api/Stocks?symbol=IBM&start=2016-02-10&end=2017-02-10**

**[api/User]**

The *UserController* has 3 overloaded methods that will return a List in XML form of the data requested by your parameters.

The format for calling this API, just like for every other method is formatted as:

**../api/User? Parameter1=value1&Parameter2=value2&…**

(continued in the same format for as many parameters as required.)

***OVERLOADS*** *–* Return List format: { DateTime date, decimal price, int count}

(string symbol, string user, string keyword)

//returns the frequency a twitter account uses a word and the stock price by day, from the date the user first used the word.

(string symbol, string user, string keyword, DateTime start)

//returns the frequency a twitter account uses a word and the stock price by day, from the date *start.*

(string symbol, string user, string keyword, DateTime start, DateTime end)

//returns the frequency a twitter account uses a word and the stock price by day, from the dates *start* to *end.*

***ALL PARAMETERS –*** *required formats:*

string symbol: must be an actual stock symbol

string user: must be an actual twitter account – just letters, no “@”

string keyword: just a simple string. There will likely be no data if the input is not a word.

string start: Keep in DateTime format = YYYY-MM-DD

string end: Keep in DateTime format = YYYY-MM-DD

**EXAMPLE CALL: ../api/User?symbol=FOX&user=realDonaldTrump&keyword=media**

**[Api/Combined]**

The *CombinedController* has 1 method that will return a List in XML form of the data requested by your parameters.

The format for calling this API, just like for every other method is formatted as:

**../api/Combined? Parameter1=value1&Parameter2=value2&…**

(continued in the same format for as many parameters as required.)

***OVERLOADS:*** Return List format: { DateTime date, decimal stockPrice, int trendingValue}

(string Hashtag, string symbol)

//returns the number of popular tweets with a given #hashtag daily and its corresponding stock price for the symbol. The time period will be the days from when the #hashtag first begins trending and ends at the last day it was trending

***ALL PARAMETERS –*** *required formats:*

string Hashtag: must be a hashtag that was trending at some point. Just the letters(no #)

string symbol: must be an actual stock symbol

**EXAMPLE CALL: ../api/Combined?Hashtag=mac&symbol=APPL**

**[Api/Twitter]**

The *TwitterController* has 1 method that will return a List in XML form of the data requested by your parameters.

The format for calling this API, just like for every other method is formatted as:

**../api/Twitter? Parameter1=value1&Parameter2=value2&…**

(continued in the same format for as many parameters as required.)

***OVERLOADS:*** Return List format: { DateTime date, int value}

(string Hashtag, string symbol)

//returns the number of popular tweets with a given #hashtag daily. The time period will be the days from when the #hashtag first begins trending and ends at the last day it was trending

***ALL PARAMETERS –*** *required formats:*

string Hashtag: must be a hashtag that was trending at some point. Just the letters(no #)

**EXAMPLE CALL: ../api/Twitter?Hashtag=google**