

# DOCUMENTATION

## [Arnav Challa]

### MockStockMarketServer

**Brief:** This is a server which simulates a stock market, with first order dependency[momentum], it has a list of stocks with some start prices, which change over time. The values themselves are arbitrary.

**Working:** The entire simulation is done inside stockSimulator.js, which is imported into the router so that it can send data responses to API requests

#### **Routes: [port 3001]**

/stocks => all stocks and their prices

/stocks/history=>all stocks and their price histories

/stocks/:symbol=>stock price of specific stock

/stocks/history/:symbol=>stock history of specific stock

Note: For this assignment, we will rely on the /stocks/history route

## **StockMarketBot**

**Brief:** This bot makes API calls to the stock server and uses the moving average crossover trading strategy to make trades.

**Working:** API calls are made to the server every second to get the updated prices, here we have arbitrarily set short average to 5s and long average to 20s. If  $SA > LA$  we buy as it indicates positive momentum and if  $SA < LA$  we sell as it indicates negative momentum.

**Routes:** [port 3000]

We have only one route that is '/' which prints the log file

**Log File:** The log file contains a record of which stock we buy, at what time and at what price. When we sell, it also records the profit/loss. When we end the session, we get the net Profit/Loss of the session.

## **RUNNING CODE:**

-Execute index.js in MockStockMarketServer folder to start the stock simulation

[optional: wait 20s]

-Execute index.js in StockMarketBot folder to start the bot

=> simply use npm start to execute each index.js

Note: As the long-period has been set to 20s, no trades will be made until 20s have passed since start of stock server

**IMPORTANT: Express is a dependency, please install it via**

*-npm i express **or** npm install*

*For any issues please contact:*

*arnav8703@gmail.com*