

**Here is the background information on your task**

You’ve been asked to assist with some development to add a chart to a trader’s dashboard allowing them to better identify under/over-valued stocks.

The trader would like to be able to monitor two historically correlated stocks and be able to visualize when the correlation between the two weakens (i.e. one stock moves proportionally more than the historical correlation would imply). This could indicate a potential trade strategy to simultaneously buy the relatively underperforming stock and sell the relatively outperforming stock. Assuming the two prices subsequently converge, the trade should be profitable.

Most data visualization for our traders is built on JPMorgan Chase's Perspective data visualization software, which is now open source. If you want to explore that, a link is provided in the resources section.

Before implementing this request using perspective, first you’ll need to interface with the relevant financial data feed and make the necessary adjustments to facilitate the monitoring of potential trade opportunities.

\* Understanding the finance and trading part is not required.

\* Being familiar with python scripting language and command line basics is not required as you will be guided in this exercise

\* (Note, you DO NOT have to install Perspective as an individual software onto your machine. All you need to complete this task is to follow the instructions in step 3)

### Here is your task

**For the first module of this project will need you to accomplish the following:**

1. Set up your system by downloading the necessary repository, files, tools and dependencies
2. Fix the broken client data feed script in the repository by making the required adjustments to it.
3. Generate a patch file of the changes you made
4. Bonus task: Add unit tests in the test script in the repository.

**We've broken down the steps for you in stages below so you can accomplish this task in an organized manner.**

**Set Up**

Before you can tackle any software or development task you need to set up your development environment. Your development environment refers to your system having all the required software installed to modify the code, as well as getting the code of the project itself onto your computer.

To do this we've created a simple PDF guide below on how to get your environment set up:

<https://cdn.theforage.com/vinternships/companyassets/Sj7temL583QAYpHXD/setup_devenv_m1_v6.pdf> (How to set up your dev environment on your machine for module one in the JPMorgan Chase Program)

<https://repl.it/@JoeFerrer/JPM-MODULE-1-REPL> (Quick start this task in your browser with Repl(Python 2.7)

<https://repl.it/@JoeFerrer/JPM-MODULE1-PY3-REPL> (Quick start this task in your browser with Repl(Python 3+)

**Making Changes**

When you’re in a work environment, you’ll usually receive tasks in the form of engineering tickets. Here is an example of what this task looks like in the form of an engineering ticket

Purpose  
We want to process the data feed of stock A and stock B’s price to enable us to analyse when trading for the stock should occur.

Acceptance Criteria

* *getDataPoint* function should return correct tuple of stock name, bid\_price, ask\_price and price. Note: price of a stock = average of bid and ask
* *getRatio* function should return the ratio of the two stock prices
* main function should output correct stock info, prices and ratio
* Upload a git patch file as the submission to this task
* Bonus: All unit tests inside client\_test.py, added/existing have to pass

To properly make the changes necessary to complete the objectives of this task, follow the guide below.

<https://cdn.theforage.com/vinternships/companyassets/Sj7temL583QAYpHXD/making_changes_m1_v4a.pdf> (Step by step guide to make code changes to complete task 1)

**Submitting Changes**

Please upload a git patch file as the submission to this task

To generate a patch file, follow the guide below.

<https://cdn.theforage.com/vinternships/companyassets/Sj7temL583QAYpHXD/create_patch_file_v3a.pdf> (How to make a patch File)

### Resources to help you with the task

Note: You do not have to use all of the resources here. Some of these resources have been mentioned in the past guides already and are just kept for your reference in case you need to go back and review them

<https://git-scm.com/book/en/v2/Getting-Started-Git-Basics> (GIT basics link)

<https://www.computerhope.com/overview.htm> (Windows command line basics)

<https://docs.python-guide.org/starting/installation/> (Installation instructions for python)

<https://pip.pypa.io/en/stable/installing/> (Installation instructions for pip)

<https://docs.python.org/2.7/library/unittest.html#assert-methods> (Guide to python unit testing and assertions)

<https://perspective.finos.org/> (JPMorgan Chase’s perspective)