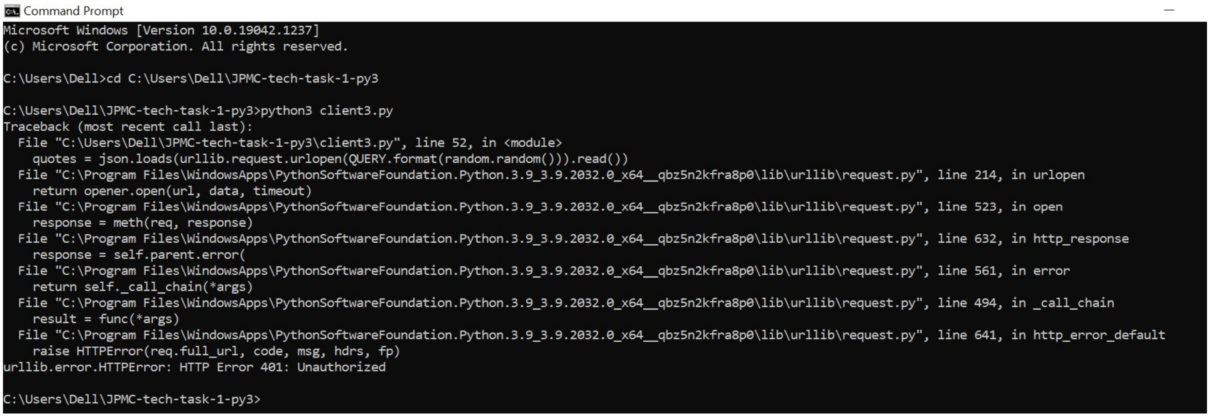
Suba Varshini Venkatesan

JPMorgan Chase & Co Software Engineering Virtual Internship

September 2021 – October 2021

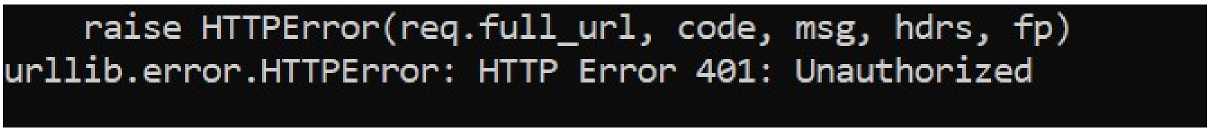
TASK 1

While executing python3 client3.py following errors occurred



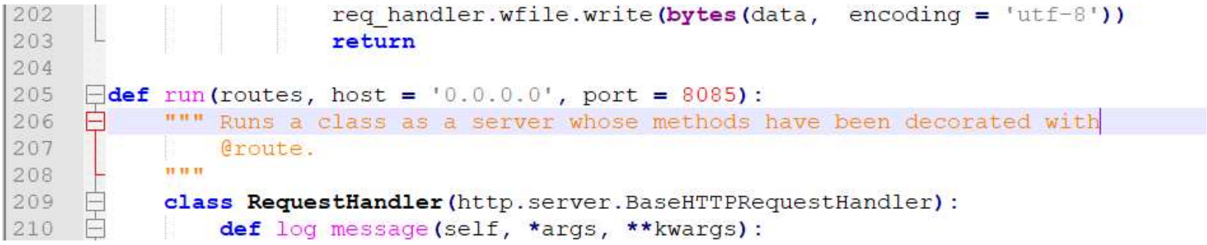
```
Microsoft Windows [Version 10.0.19042.1237]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Dell>cd C:\Users\Dell\JPMC-tech-task-1-py3

C:\Users\Dell\JPMC-tech-task-1-py3>python3 client3.py
Traceback (most recent call last):
  File "C:\Users\Dell\JPMC-tech-task-1-py3\client3.py", line 52, in <module>
    quotes = json.loads(urllib.request.urlopen(QUERY.format(random.random())).read())
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.2032.0_x64__qbz5n2kfra8p0\lib\urllib\request.py", line 214, in urlopen
    return opener.open(url, data, timeout)
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.2032.0_x64__qbz5n2kfra8p0\lib\urllib\request.py", line 523, in open
    response = meth(req, response)
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.2032.0_x64__qbz5n2kfra8p0\lib\urllib\request.py", line 632, in http_response
    response = self.parent.error(
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.2032.0_x64__qbz5n2kfra8p0\lib\urllib\request.py", line 561, in error
    return self._call_chain(*args)
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.2032.0_x64__qbz5n2kfra8p0\lib\urllib\request.py", line 494, in _call_chain
    result = func(*args)
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.2032.0_x64__qbz5n2kfra8p0\lib\urllib\request.py", line 641, in http_error_default
    raise HTTPError(req.full_url, code, msg, hdrs, fp)
urllib.error.HTTPError: HTTP Error 401: Unauthorized

C:\Users\Dell\JPMC-tech-task-1-py3>
```

Specific error:



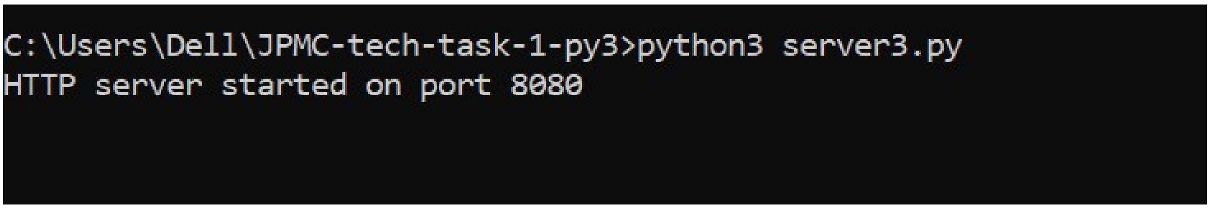
```
    raise HTTPError(req.full_url, code, msg, hdrs, fp)
urllib.error.HTTPError: HTTP Error 401: Unauthorized
```

How I resolved:



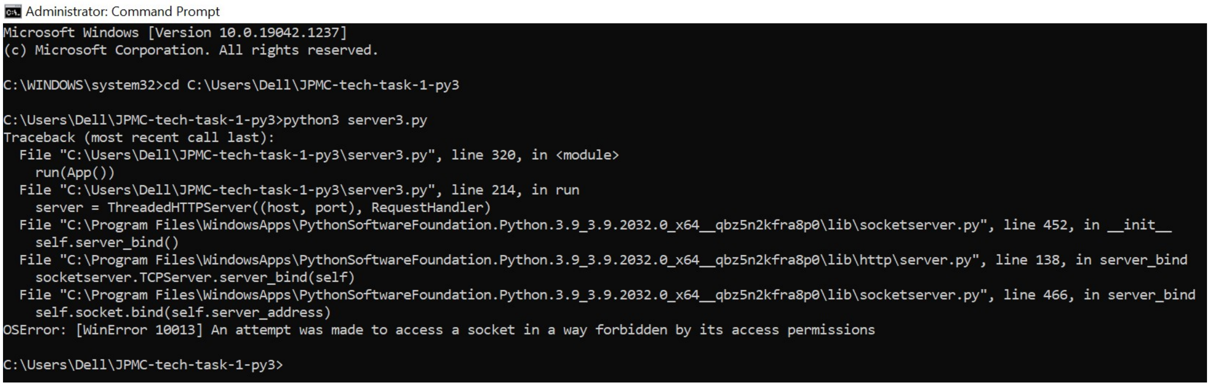
```
                req_handler.wfile.write(bytes(data,  encoding = 'utf-8'))
                return

def run(routes, host = '0.0.0.0', port = 8085):
    """ Runs a class as a server whose methods have been decorated with
        @route.
    """
    class RequestHandler(http.server.BaseHTTPRequestHandler):
        def log_message(self, *args, **kwargs):
```

Change the port number from 8080 to 8085



```
C:\Users\Dell\JPMC-tech-task-1-py3>python3 server3.py
HTTP server started on port 8080
```

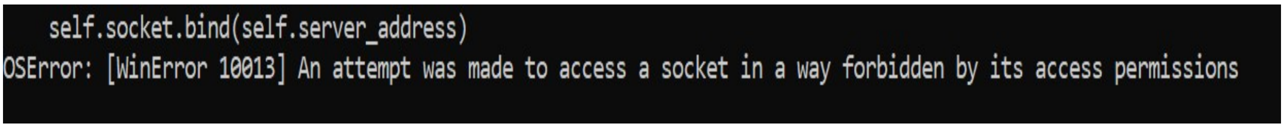
While executing python3 server3.py in Admin mode in cmd following errors occurred



Administrator: Command Prompt

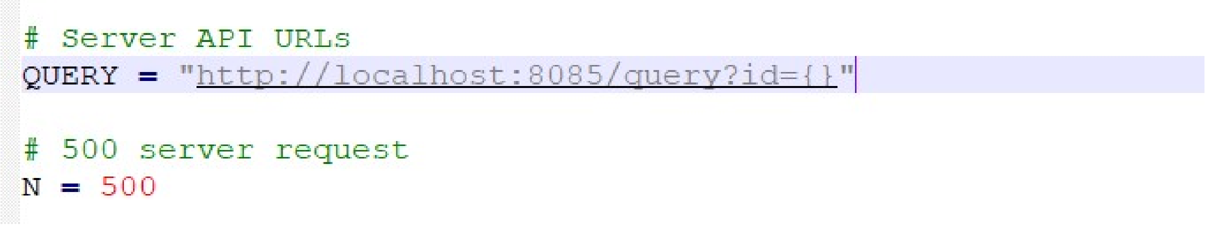
```
Microsoft Windows [Version 10.0.19042.1237]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>cd C:\Users\Dell\JPMC-tech-task-1-py3

C:\Users\Dell\JPMC-tech-task-1-py3>python3 server3.py
Traceback (most recent call last):
  File "C:\Users\Dell\JPMC-tech-task-1-py3\server3.py", line 320, in <module>
    run(App())
  File "C:\Users\Dell\JPMC-tech-task-1-py3\server3.py", line 214, in run
    server = ThreadedHTTPServer((host, port), RequestHandler)
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.2032.0_x64__qbz5n2kfra8p0\lib\socketserver.py", line 452, in __init__
    self.server_bind()
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.2032.0_x64__qbz5n2kfra8p0\lib\http\server.py", line 138, in server_bind
    socketserver.TCPServer.server_bind(self)
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.2032.0_x64__qbz5n2kfra8p0\lib\socketserver.py", line 466, in server_bind
    self.socket.bind(self.server_address)
OSError: [WinError 10013] An attempt was made to access a socket in a way forbidden by its access permissions

C:\Users\Dell\JPMC-tech-task-1-py3>
```

Specific error



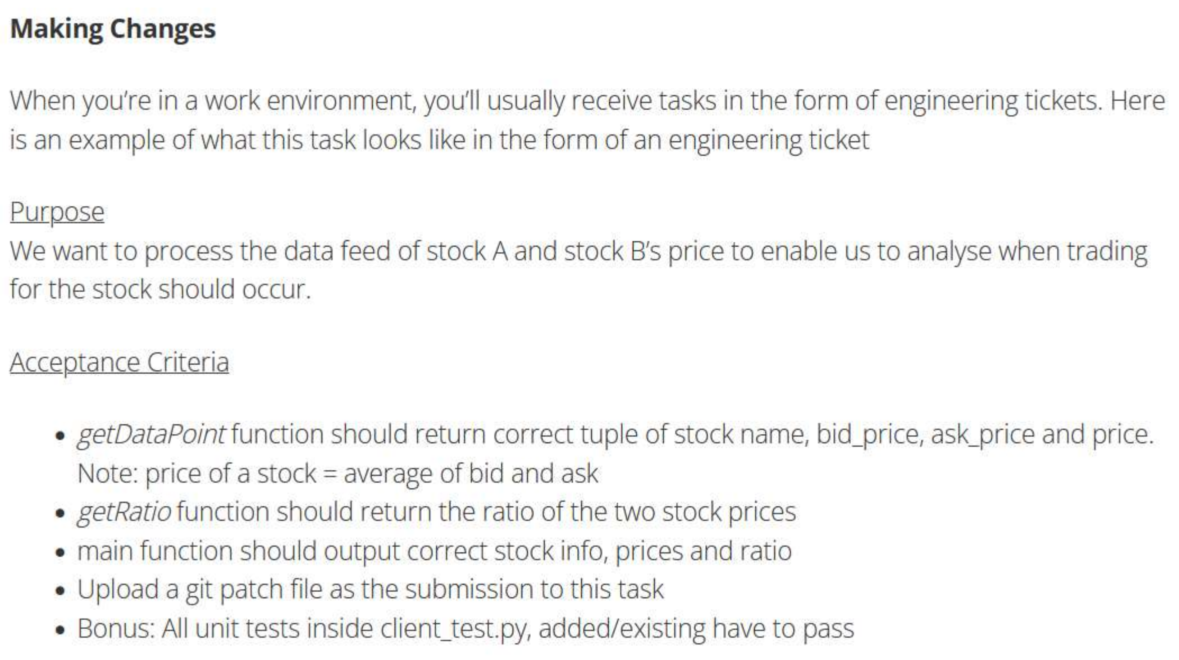
```
    self.socket.bind(self.server_address)
OSError: [WinError 10013] An attempt was made to access a socket in a way forbidden by its access permissions
```

Change port to 8085 in client3.py



```
# Server API URLs
QUERY = "http://localhost:8085/query?id={}"

# 500 server request
N = 500
```

After running the commands respectively starting the task:



Making Changes

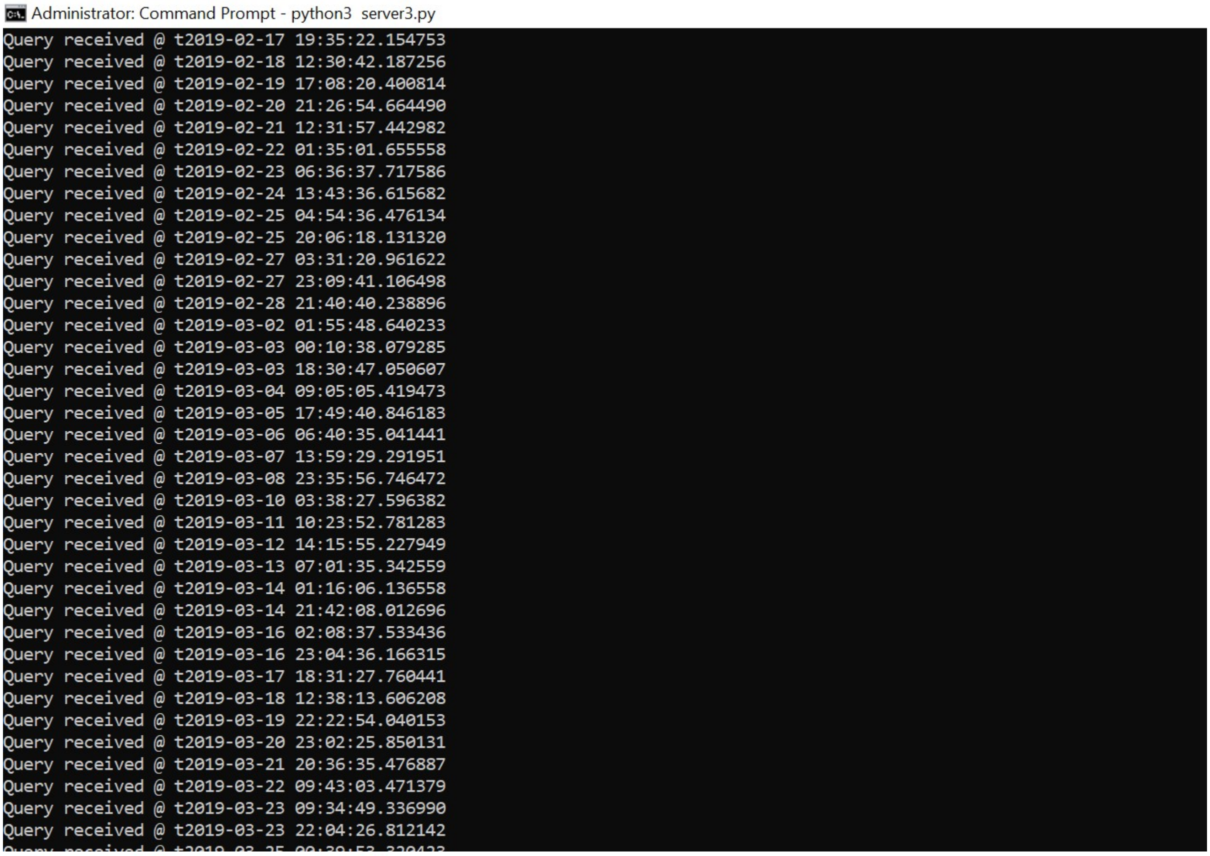
When you're in a work environment, you'll usually receive tasks in the form of engineering tickets. Here is an example of what this task looks like in the form of an engineering ticket

Purpose

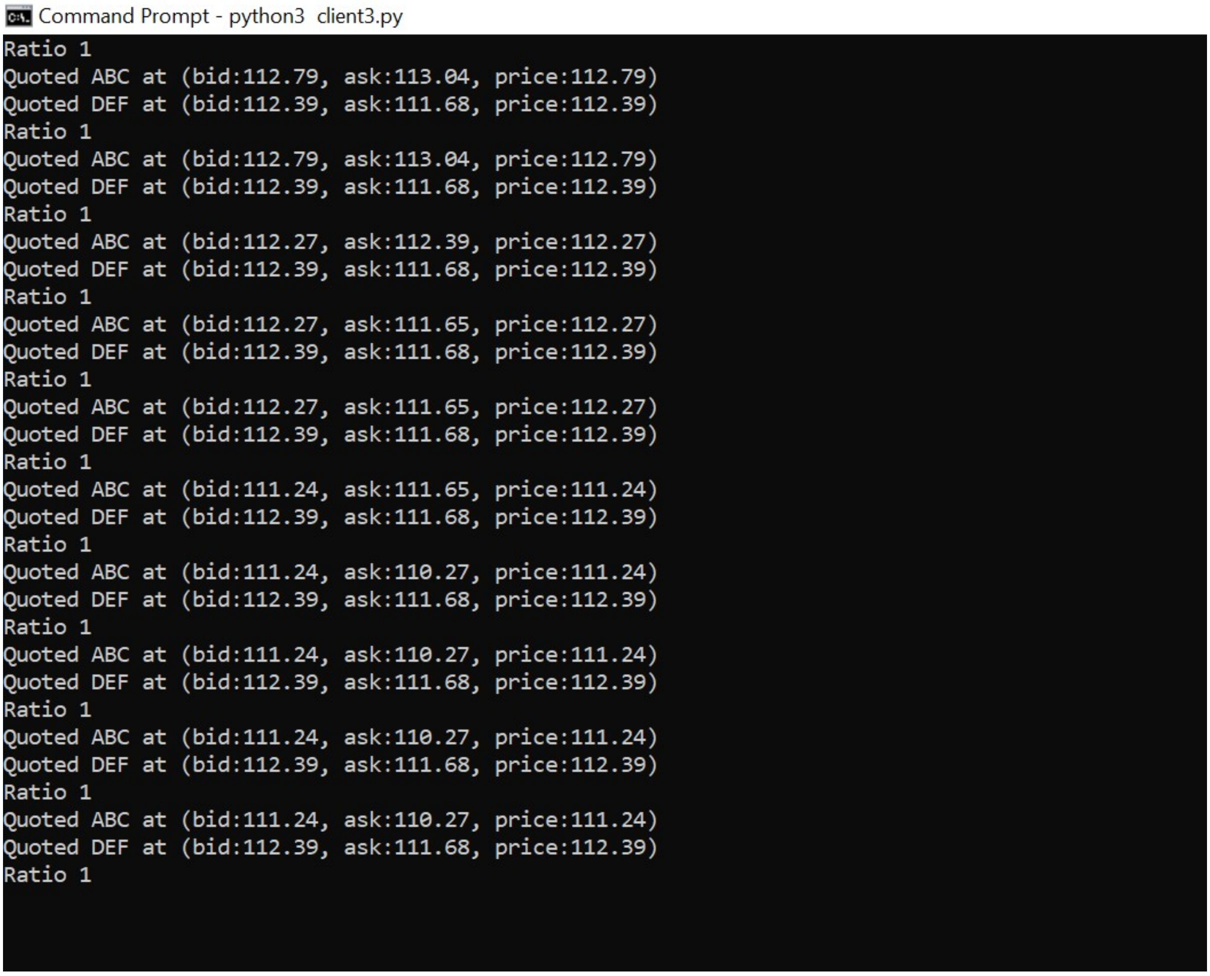
We want to process the data feed of stock A and stock B's price to enable us to analyse when trading for the stock should occur.

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

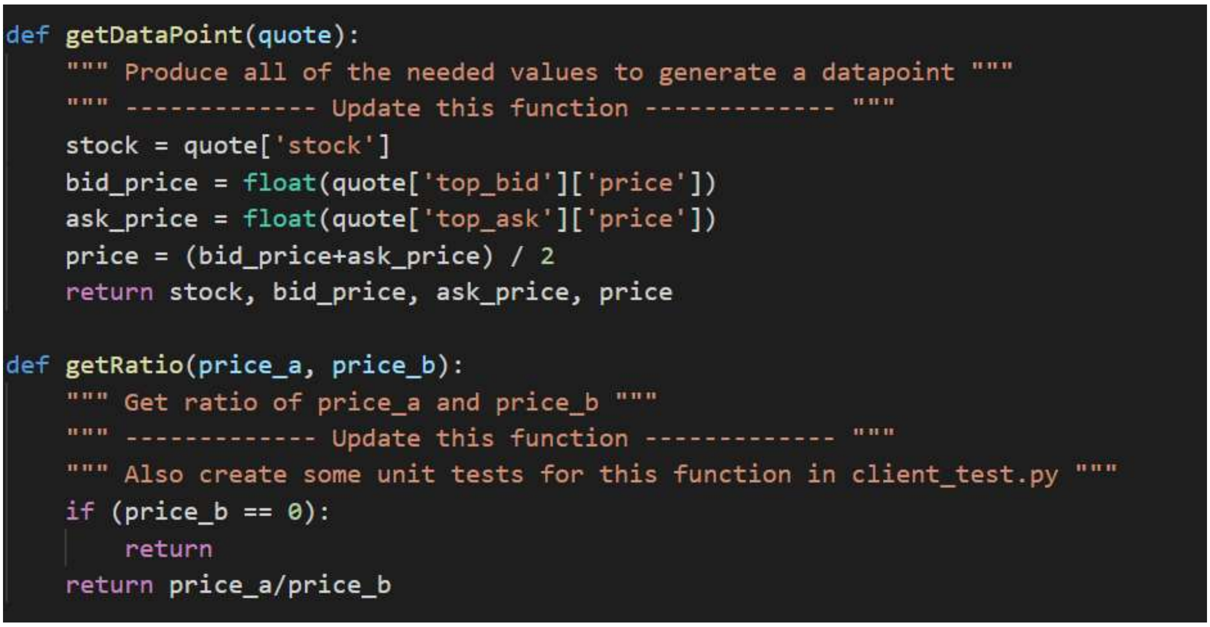


```
Query received @ t2019-02-17 19:35:22.154753
Query received @ t2019-02-18 12:30:42.187256
Query received @ t2019-02-19 17:08:20.400814
Query received @ t2019-02-20 21:26:54.664490
Query received @ t2019-02-21 12:31:57.442982
Query received @ t2019-02-22 01:35:01.655558
Query received @ t2019-02-23 06:36:37.717586
Query received @ t2019-02-24 13:43:36.615682
Query received @ t2019-02-25 04:54:36.476134
Query received @ t2019-02-25 20:06:18.131320
Query received @ t2019-02-27 03:31:20.961622
Query received @ t2019-02-27 23:09:41.106498
Query received @ t2019-02-28 21:40:40.238896
Query received @ t2019-03-02 01:55:48.640233
Query received @ t2019-03-03 00:10:38.079285
Query received @ t2019-03-03 18:30:47.050607
Query received @ t2019-03-04 09:05:05.419473
Query received @ t2019-03-05 17:49:40.846183
Query received @ t2019-03-06 06:40:35.041441
Query received @ t2019-03-07 13:59:29.291951
Query received @ t2019-03-08 23:35:56.746472
Query received @ t2019-03-10 03:38:27.596382
Query received @ t2019-03-11 10:23:52.781283
Query received @ t2019-03-12 14:15:55.227949
Query received @ t2019-03-13 07:01:35.342559
Query received @ t2019-03-14 01:16:06.136558
Query received @ t2019-03-14 21:42:08.012696
Query received @ t2019-03-16 02:08:37.533436
Query received @ t2019-03-16 23:04:36.166315
Query received @ t2019-03-17 18:31:27.760441
Query received @ t2019-03-18 12:38:13.606208
Query received @ t2019-03-19 22:22:54.040153
Query received @ t2019-03-20 23:02:25.850131
Query received @ t2019-03-21 20:36:35.476887
Query received @ t2019-03-22 09:43:03.471379
Query received @ t2019-03-23 09:34:49.336990
Query received @ t2019-03-23 22:04:26.812142
Query received @ t2019-03-25 00:39:53.320423
```

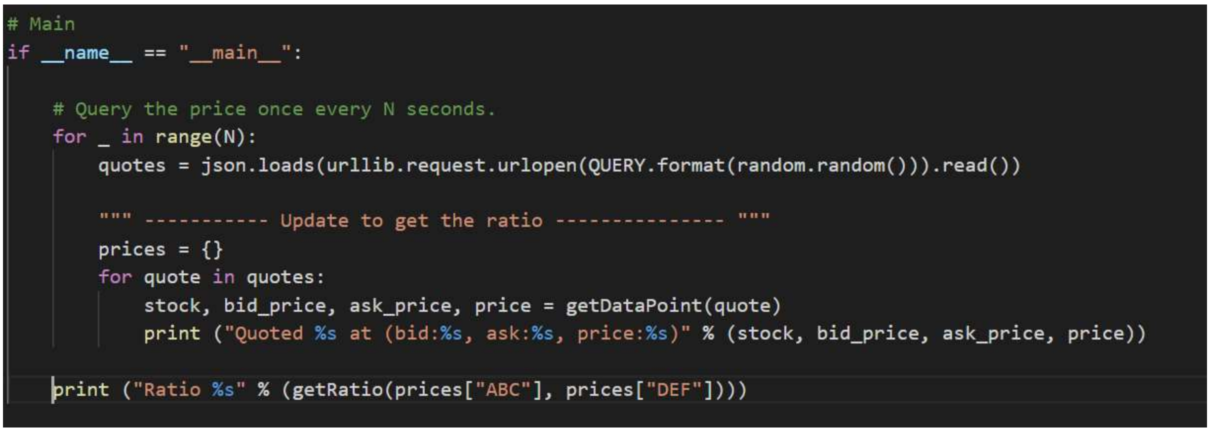


```
Ratio 1
Quoted ABC at (bid:112.79, ask:113.04, price:112.79)
Quoted DEF at (bid:112.39, ask:111.68, price:112.39)
Ratio 1
Quoted ABC at (bid:112.79, ask:113.04, price:112.79)
Quoted DEF at (bid:112.39, ask:111.68, price:112.39)
Ratio 1
Quoted ABC at (bid:112.27, ask:112.39, price:112.27)
Quoted DEF at (bid:112.39, ask:111.68, price:112.39)
Ratio 1
Quoted ABC at (bid:112.27, ask:111.65, price:112.27)
Quoted DEF at (bid:112.39, ask:111.68, price:112.39)
Ratio 1
Quoted ABC at (bid:112.27, ask:111.65, price:112.27)
Quoted DEF at (bid:112.39, ask:111.68, price:112.39)
Ratio 1
Quoted ABC at (bid:111.24, ask:111.65, price:111.24)
Quoted DEF at (bid:112.39, ask:111.68, price:112.39)
Ratio 1
Quoted ABC at (bid:111.24, ask:110.27, price:111.24)
Quoted DEF at (bid:112.39, ask:111.68, price:112.39)
Ratio 1
Quoted ABC at (bid:111.24, ask:110.27, price:111.24)
Quoted DEF at (bid:112.39, ask:111.68, price:112.39)
Ratio 1
Quoted ABC at (bid:111.24, ask:110.27, price:111.24)
Quoted DEF at (bid:112.39, ask:111.68, price:112.39)
Ratio 1
Quoted ABC at (bid:111.24, ask:110.27, price:111.24)
Quoted DEF at (bid:112.39, ask:111.68, price:112.39)
Ratio 1
```

Changing the functions:

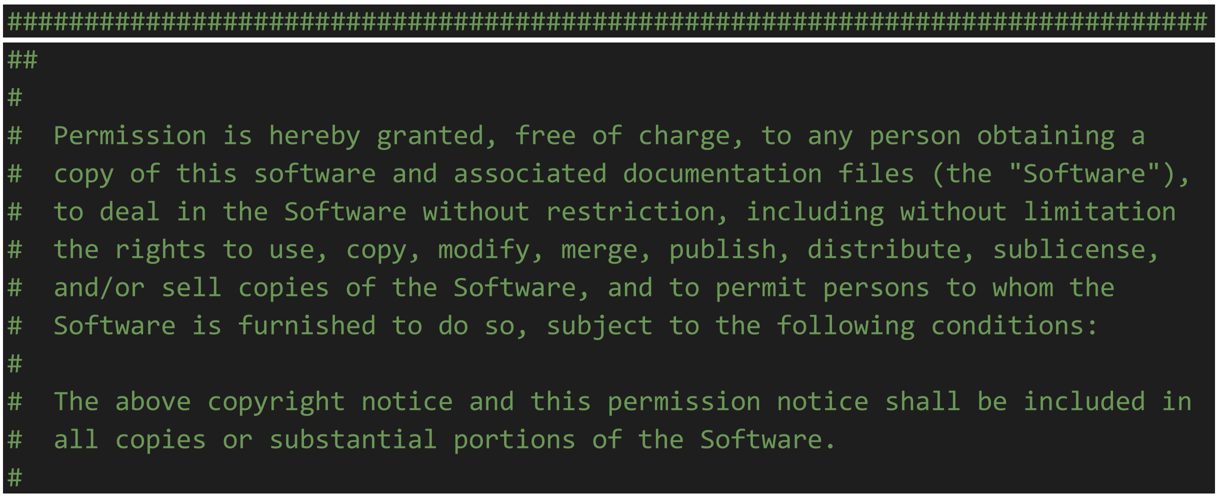


```
def getDataPoint(quote):
    """ Produce all of the needed values to generate a datapoint """
    """ ------------- Update this function ------------- """
    stock = quote['stock']
    bid_price = float(quote['top_bid']['price'])
    ask_price = float(quote['top_ask']['price'])
    price = (bid_price+ask_price) / 2
    return stock, bid_price, ask_price, price

def getRatio(price_a, price_b):
    """ Get ratio of price_a and price_b """
    """ ------------- Update this function ------------- """
    """ Also create some unit tests for this function in client_test.py """
    if (price_b == 0):
        return
    return price_a/price_b
```

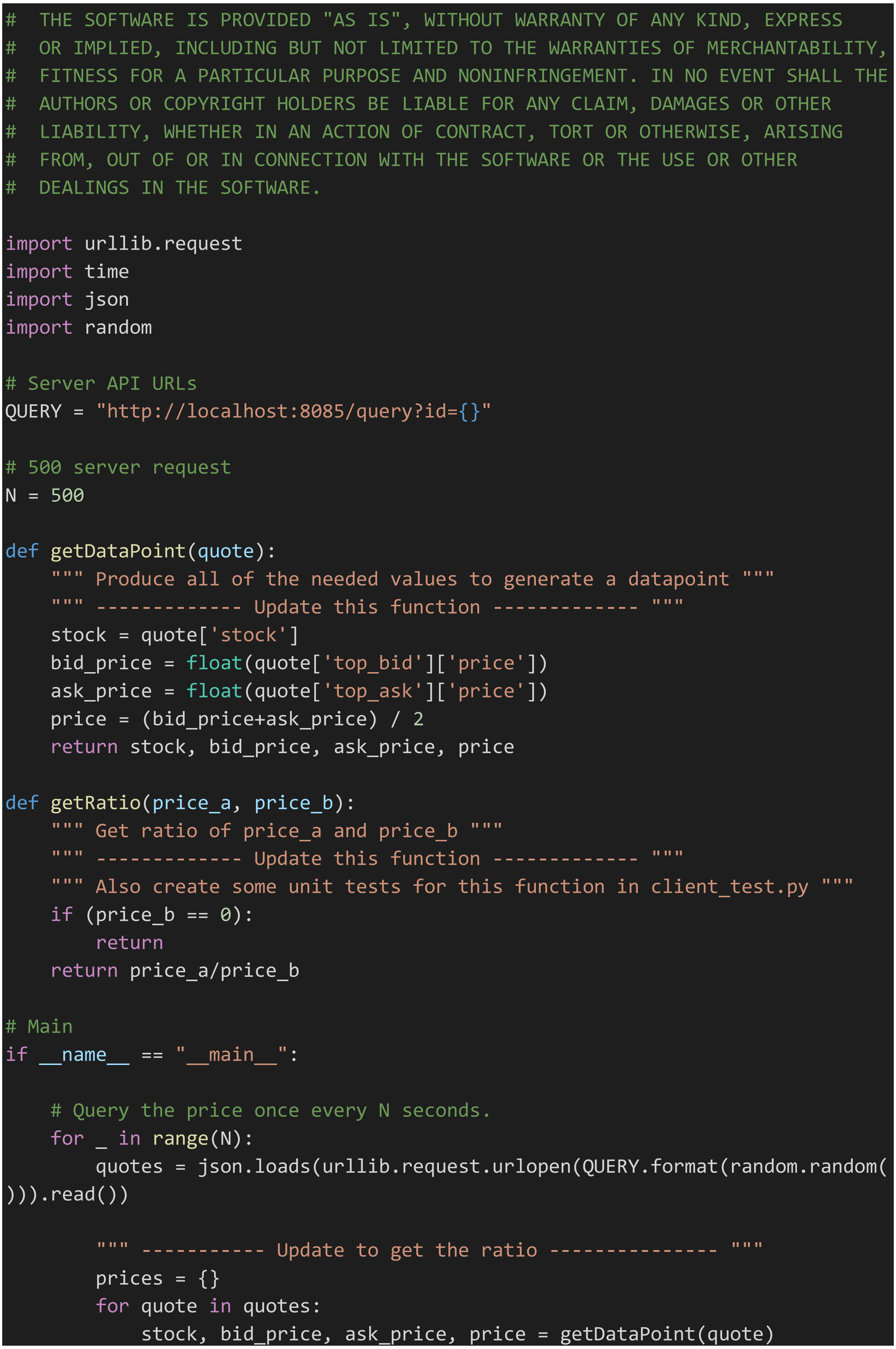


```
# Main
if __name__ == "__main__":

    # Query the price once every N seconds.
    for _ in range(N):
        quotes = json.loads(urllib.request.urlopen(QUERY.format(random.random())).read())

        """ ----------- Update to get the ratio --------------- """
        prices = {}
        for quote in quotes:
            stock, bid_price, ask_price, price = getDataPoint(quote)
            print ("Quoted %s at (bid:%s, ask:%s, price:%s)" % (stock, bid_price, ask_price, price))

    print ("Ratio %s" % (getRatio(prices["ABC"], prices["DEF"])))
```

Code:



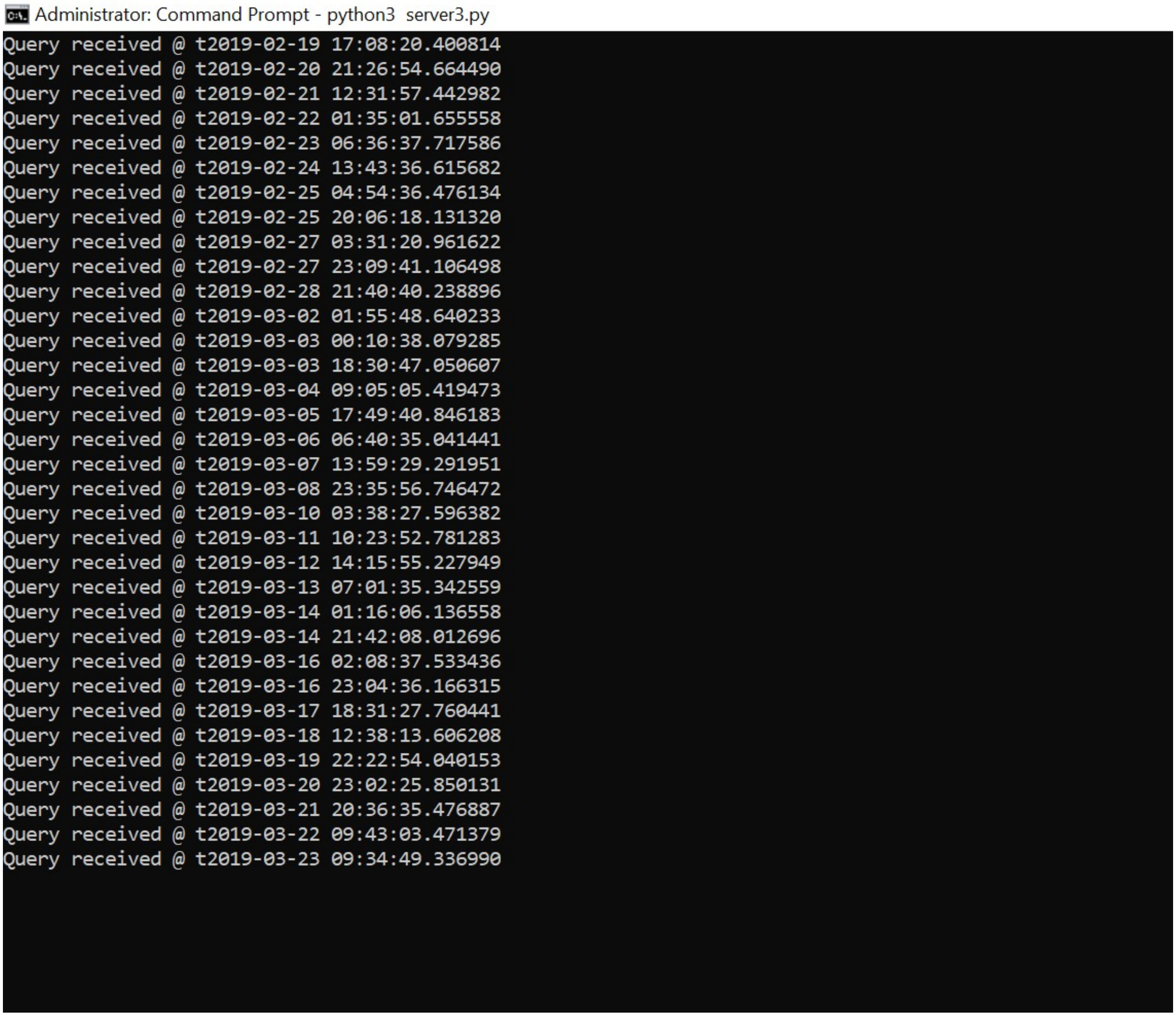
```
##############################################################################
##
#
#  Permission is hereby granted, free of charge, to any person obtaining a
#  copy of this software and associated documentation files (the "Software"),
#  to deal in the Software without restriction, including without limitation
#  the rights to use, copy, modify, merge, publish, distribute, sublicense,
#  and/or sell copies of the Software, and to permit persons to whom the
#  Software is furnished to do so, subject to the following conditions:
#
#  The above copyright notice and this permission notice shall be included in
#  all copies or substantial portions of the Software.
#
```



```
#  THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS
#  OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY,
#  FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE
#  AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER
#  LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING
#  FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER
#  DEALINGS IN THE SOFTWARE.

import urllib.request
import time
import json
import random

# Server API URLs
QUERY = "http://localhost:8085/query?id={}"

# 500 server request
N = 500

def getDataPoint(quote):
    """ Produce all of the needed values to generate a datapoint """
    """ ------------- Update this function ------------- """
    stock = quote['stock']
    bid_price = float(quote['top_bid']['price'])
    ask_price = float(quote['top_ask']['price'])
    price = (bid_price+ask_price) / 2
    return stock, bid_price, ask_price, price

def getRatio(price_a, price_b):
    """ Get ratio of price_a and price_b """
    """ ------------- Update this function ------------- """
    """ Also create some unit tests for this function in client_test.py """
    if (price_b == 0):
        return
    return price_a/price_b

# Main
if __name__ == "__main__":

    # Query the price once every N seconds.
    for _ in range(N):
        quotes = json.loads(urllib.request.urlopen(QUERY.format(random.random(
))).read())

        """ ----------- Update to get the ratio --------------- """
        prices = {}
        for quote in quotes:
            stock, bid_price, ask_price, price = getDataPoint(quote)
            print ("Quoted %s at (bid:%s, ask:%s, price:%s)" % (stock, bid_pri
ce, ask_price, price))

    print ("Ratio %s" % (getRatio(prices["ABC"], prices["DEF"])))
```

Output:

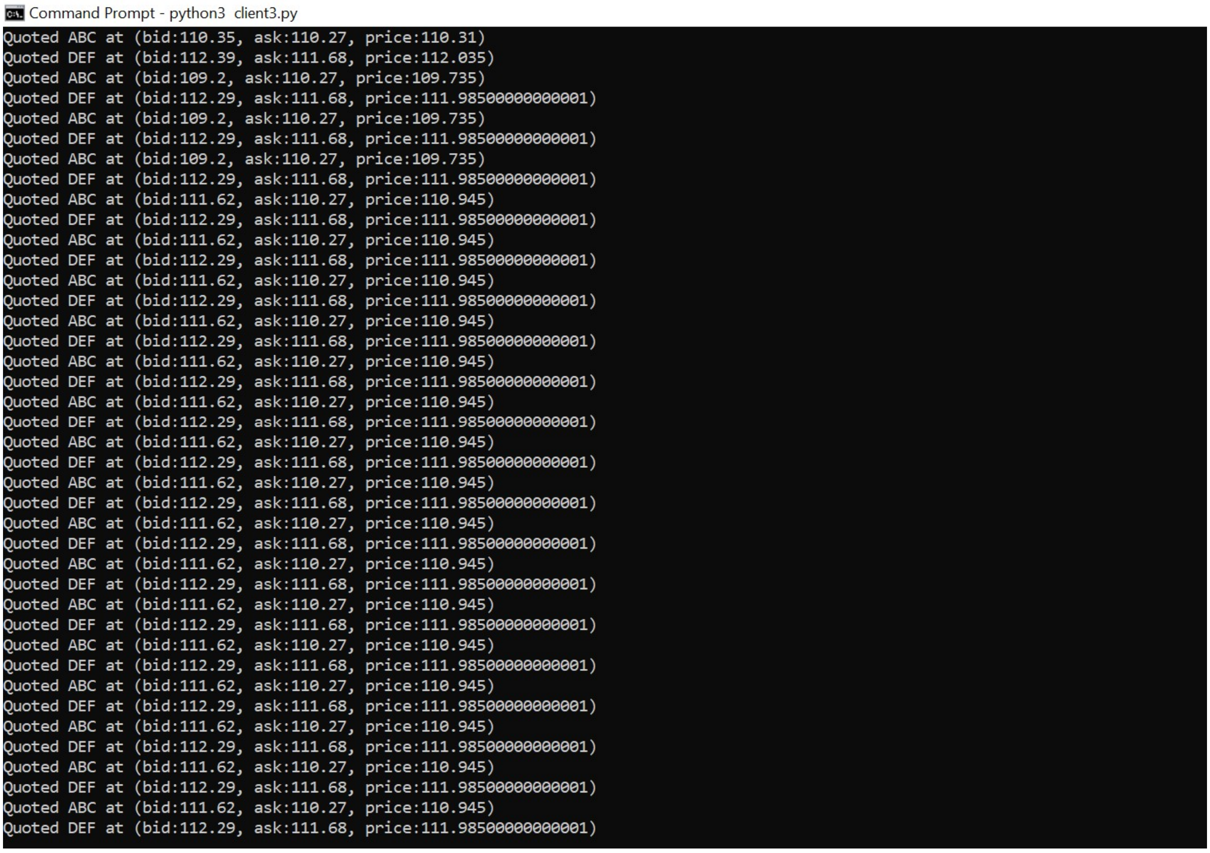
From server3.py



Administrator: Command Prompt - python3 server3.py

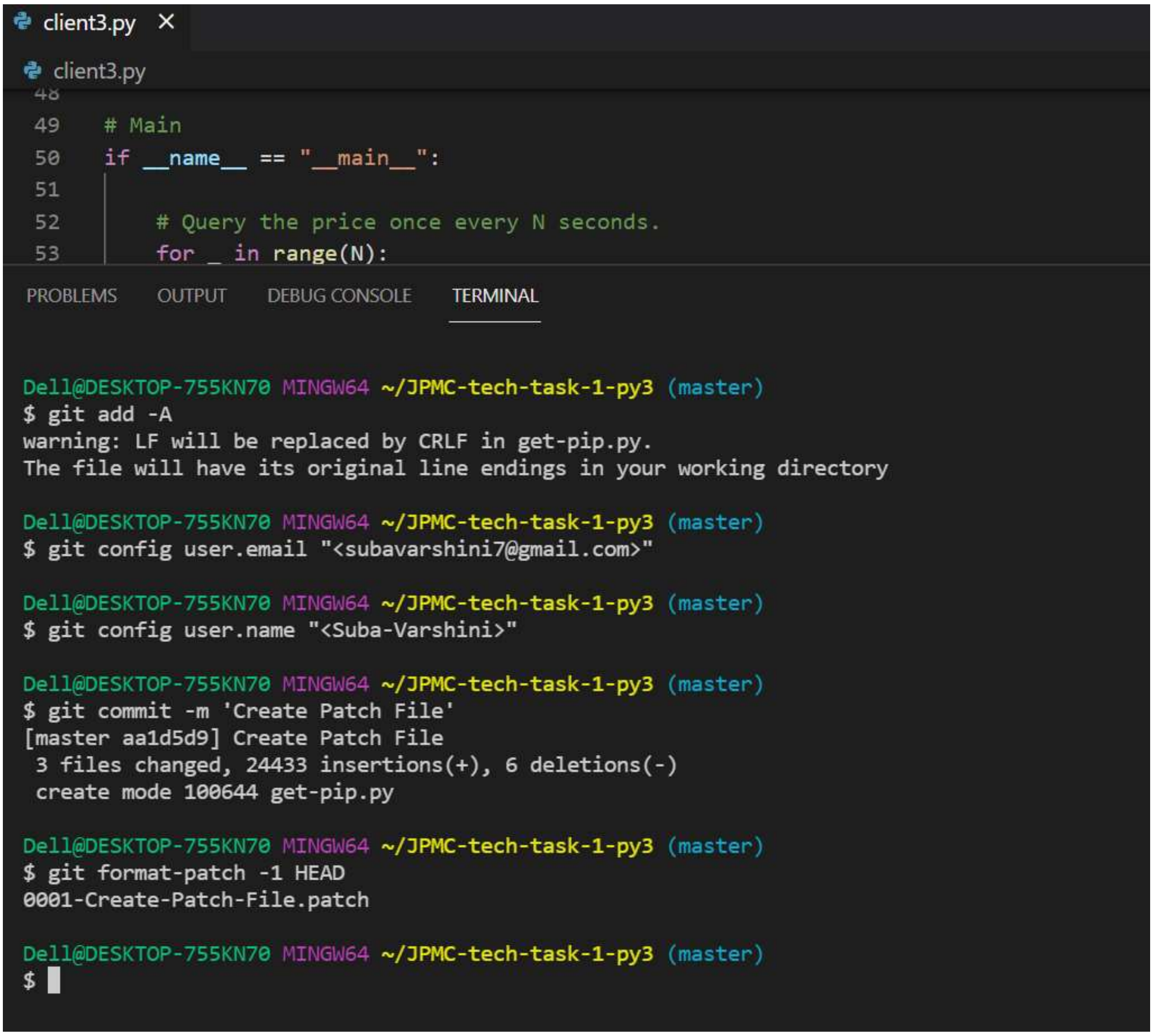
```
Query received @ t2019-02-19 17:08:20.400814
Query received @ t2019-02-20 21:26:54.664490
Query received @ t2019-02-21 12:31:57.442982
Query received @ t2019-02-22 01:35:01.655558
Query received @ t2019-02-23 06:36:37.717586
Query received @ t2019-02-24 13:43:36.615682
Query received @ t2019-02-25 04:54:36.476134
Query received @ t2019-02-25 20:06:18.131320
Query received @ t2019-02-27 03:31:20.961622
Query received @ t2019-02-27 23:09:41.106498
Query received @ t2019-02-28 21:40:40.238896
Query received @ t2019-03-02 01:55:48.640233
Query received @ t2019-03-03 00:10:38.079285
Query received @ t2019-03-03 18:30:47.050607
Query received @ t2019-03-04 09:05:05.419473
Query received @ t2019-03-05 17:49:40.846183
Query received @ t2019-03-06 06:40:35.041441
Query received @ t2019-03-07 13:59:29.291951
Query received @ t2019-03-08 23:35:56.746472
Query received @ t2019-03-10 03:38:27.596382
Query received @ t2019-03-11 10:23:52.781283
Query received @ t2019-03-12 14:15:55.227949
Query received @ t2019-03-13 07:01:35.342559
Query received @ t2019-03-14 01:16:06.136558
Query received @ t2019-03-14 21:42:08.012696
Query received @ t2019-03-16 02:08:37.533436
Query received @ t2019-03-16 23:04:36.166315
Query received @ t2019-03-17 18:31:27.760441
Query received @ t2019-03-18 12:38:13.606208
Query received @ t2019-03-19 22:22:54.040153
Query received @ t2019-03-20 23:02:25.850131
Query received @ t2019-03-21 20:36:35.476887
Query received @ t2019-03-22 09:43:03.471379
Query received @ t2019-03-23 09:34:49.336990
```

From client3.py



```
Quoted ABC at (bid:110.35, ask:110.27, price:110.31)
Quoted DEF at (bid:112.39, ask:111.68, price:112.035)
Quoted ABC at (bid:109.2, ask:110.27, price:109.735)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:109.2, ask:110.27, price:109.735)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:109.2, ask:110.27, price:109.735)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
Quoted ABC at (bid:111.62, ask:110.27, price:110.945)
Quoted DEF at (bid:112.29, ask:111.68, price:111.98500000000001)
```

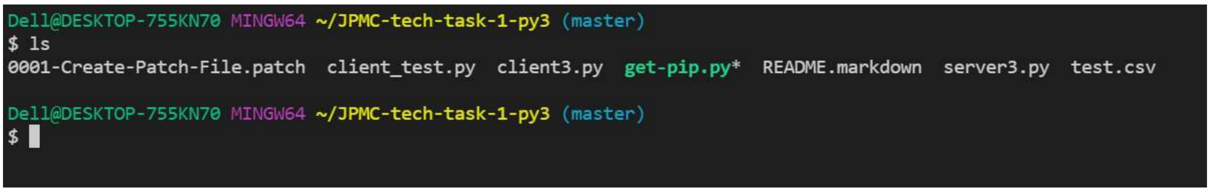
Patch File created:



```
# Main
if __name__ == "__main__":

    # Query the price once every N seconds.
    for _ in range(N):
```

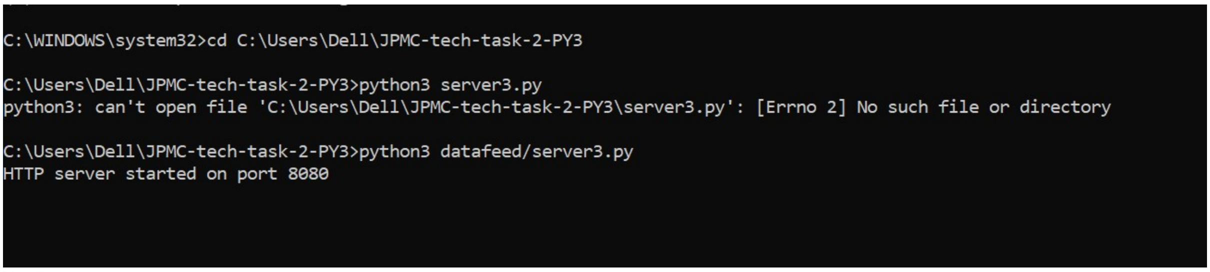
```
Dell@DESKTOP-755KN70 MINGW64 ~/JPMC-tech-task-1-py3 (master)
$ git add -A
warning: LF will be replaced by CRLF in get-pip.py.
The file will have its original line endings in your working directory

Dell@DESKTOP-755KN70 MINGW64 ~/JPMC-tech-task-1-py3 (master)
$ git config user.email "<subavarshini7@gmail.com>"

Dell@DESKTOP-755KN70 MINGW64 ~/JPMC-tech-task-1-py3 (master)
$ git config user.name "<Suba-Varshini>"

Dell@DESKTOP-755KN70 MINGW64 ~/JPMC-tech-task-1-py3 (master)
$ git commit -m 'Create Patch File'
[master aa1d5d9] Create Patch File
 3 files changed, 24433 insertions(+), 6 deletions(-)
 create mode 100644 get-pip.py

Dell@DESKTOP-755KN70 MINGW64 ~/JPMC-tech-task-1-py3 (master)
$ git format-patch -1 HEAD
0001-Create-Patch-File.patch

Dell@DESKTOP-755KN70 MINGW64 ~/JPMC-tech-task-1-py3 (master)
$ 
```

Using ls command:

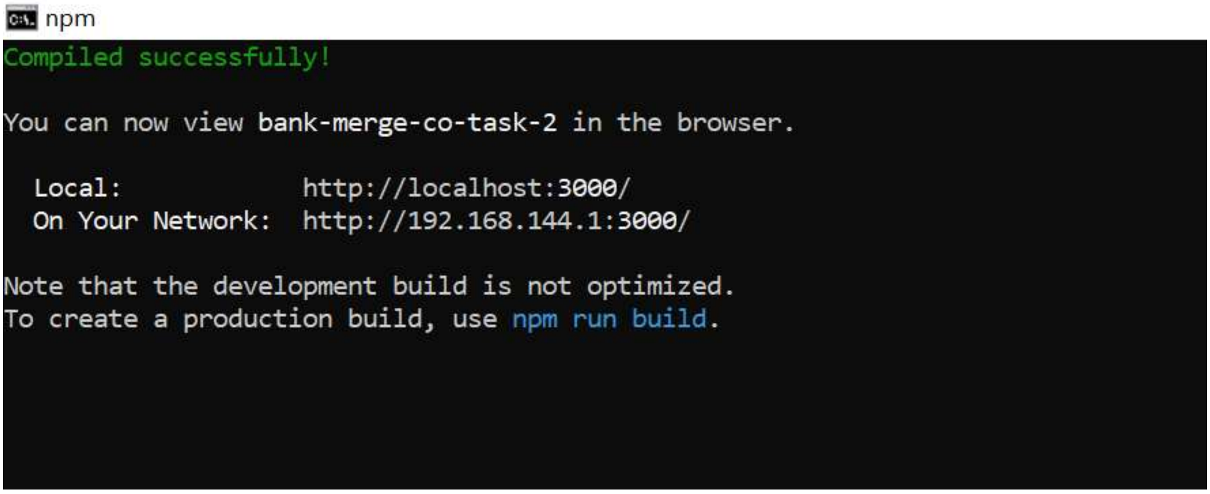


```
Dell@DESKTOP-755KN70 MINGW64 ~/JPMC-tech-task-1-py3 (master)
$ ls
0001-Create-Patch-File.patch  client_test.py  client3.py  get-pip.py*  README.markdown  server3.py  test.csv

Dell@DESKTOP-755KN70 MINGW64 ~/JPMC-tech-task-1-py3 (master)
$ 
```

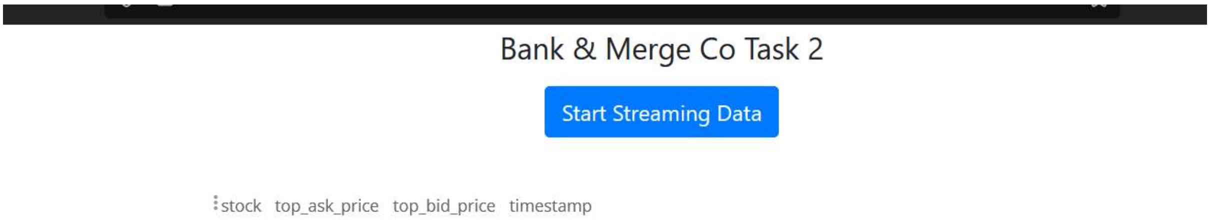
TASK 2



```
C:\WINDOWS\system32>cd C:\Users\Dell\JPMC-tech-task-2-PY3

C:\Users\Dell\JPMC-tech-task-2-PY3>python3 server3.py
python3: can't open file 'C:\Users\Dell\JPMC-tech-task-2-PY3\server3.py': [Errno 2] No such file or directory

C:\Users\Dell\JPMC-tech-task-2-PY3>python3 datafeed/server3.py
HTTP server started on port 8080
```

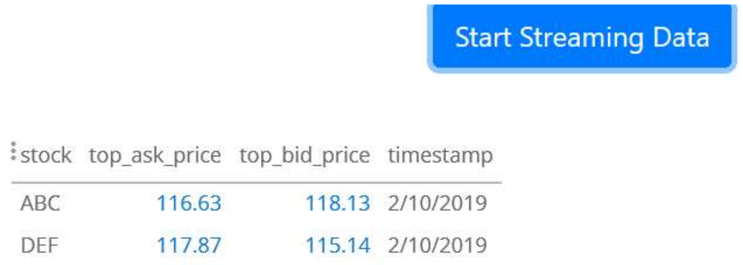


```
Compiled successfully!

You can now view bank-merge-co-task-2 in the browser.

  Local:            http://localhost:3000/
  On Your Network:  http://192.168.144.1:3000/

Note that the development build is not optimized.
To create a production build, use npm run build.
```

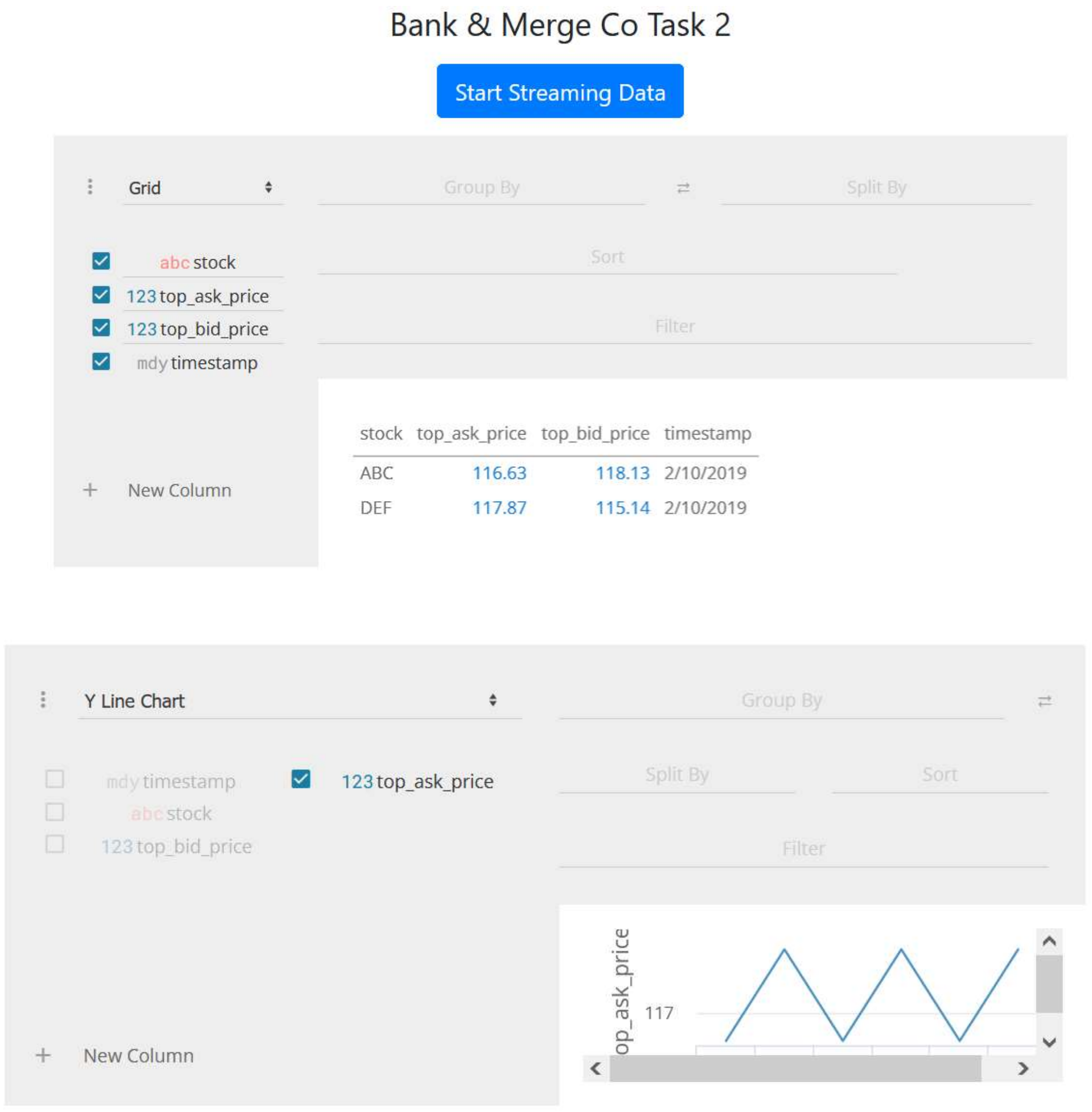


Bank & Merge Co Task 2

stock	top_ask_price	top_bid_price	timestamp



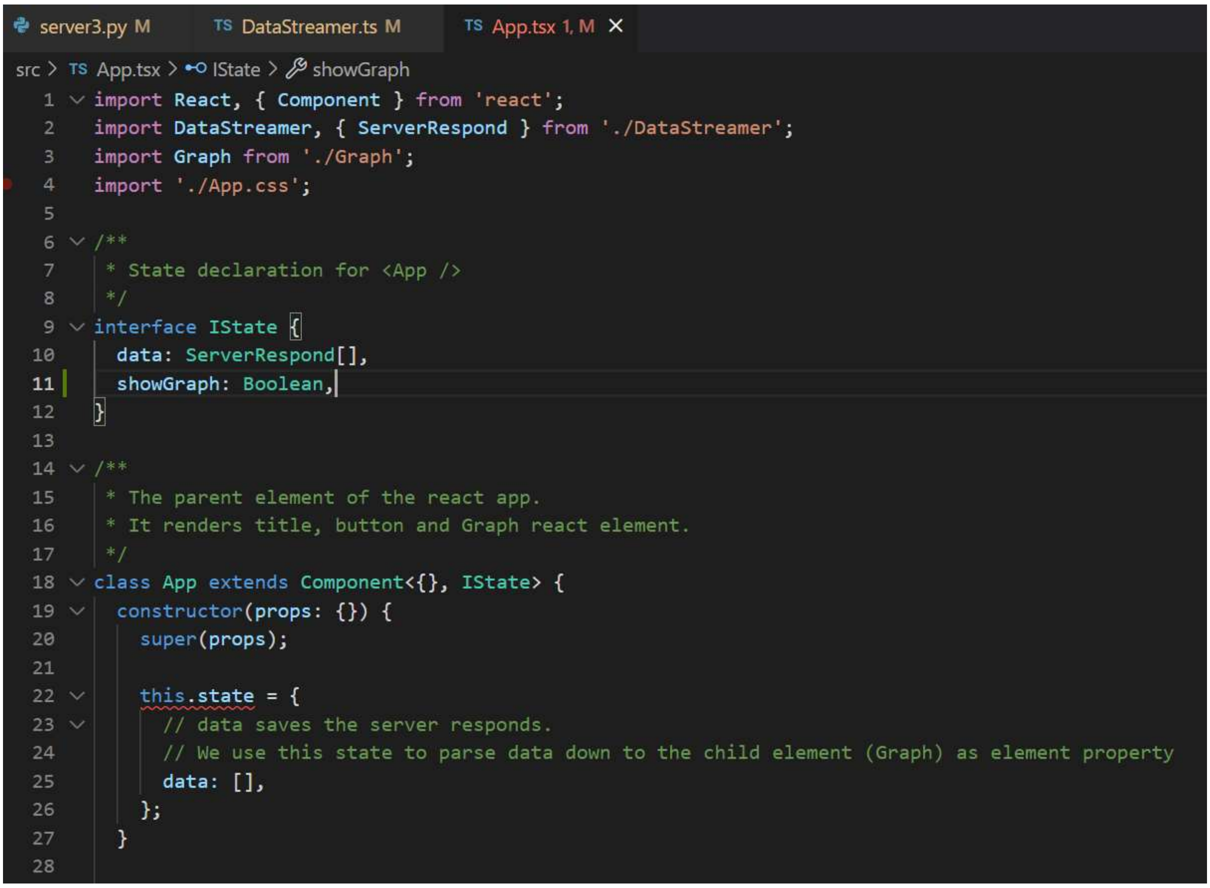
stock	top_ask_price	top_bid_price	timestamp
ABC	116.63	118.13	2/10/2019
DEF	117.87	115.14	2/10/2019



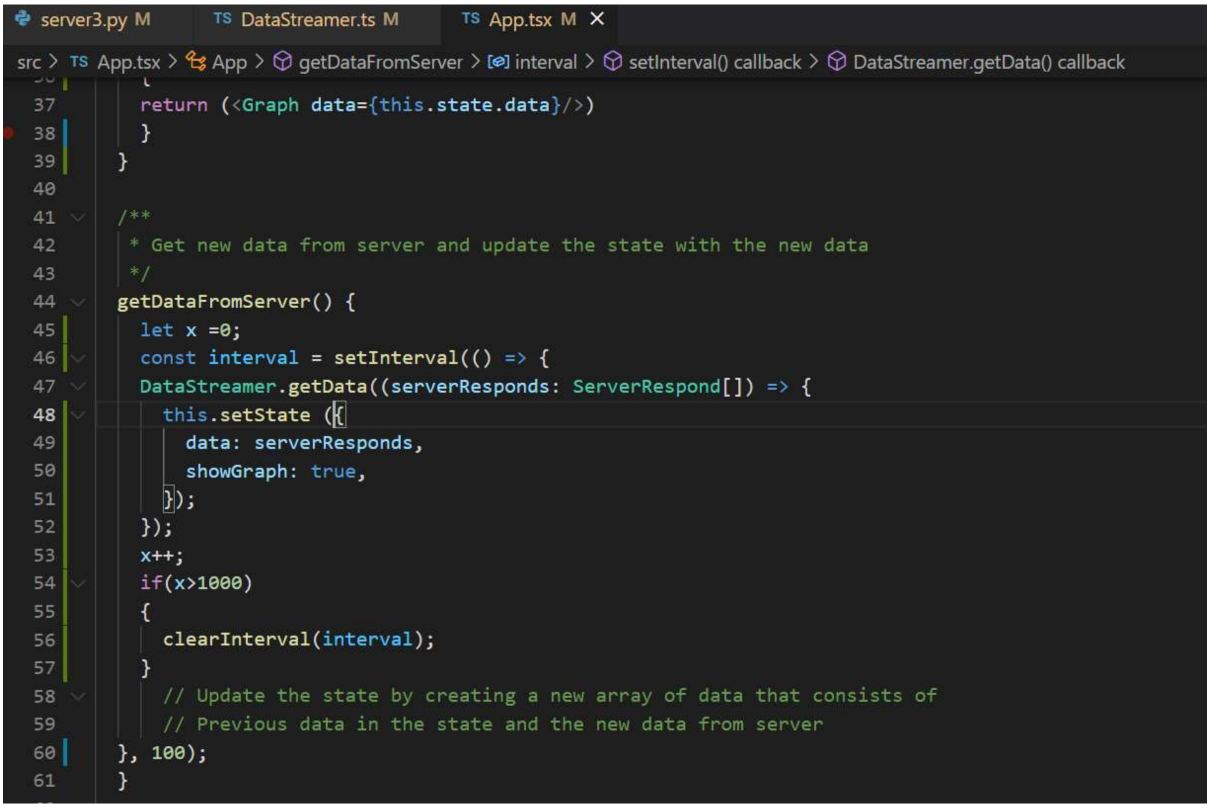
Bank & Merge Co Task 2

stock	top_ask_price	top_bid_price	timestamp
ABC	116.63	118.13	2/10/2019
DEF	117.87	115.14	2/10/2019

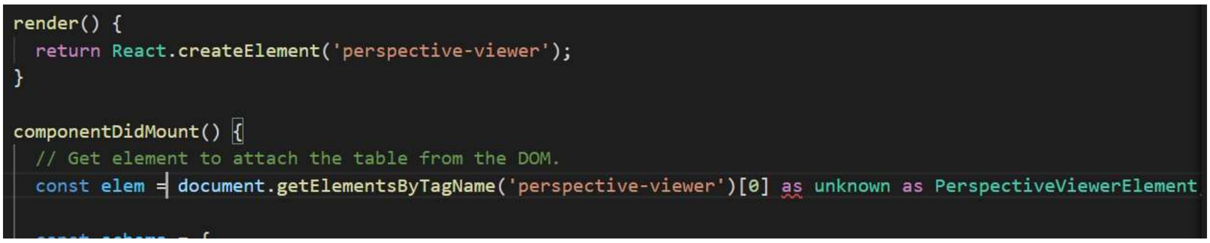
Making changes in App.tsx



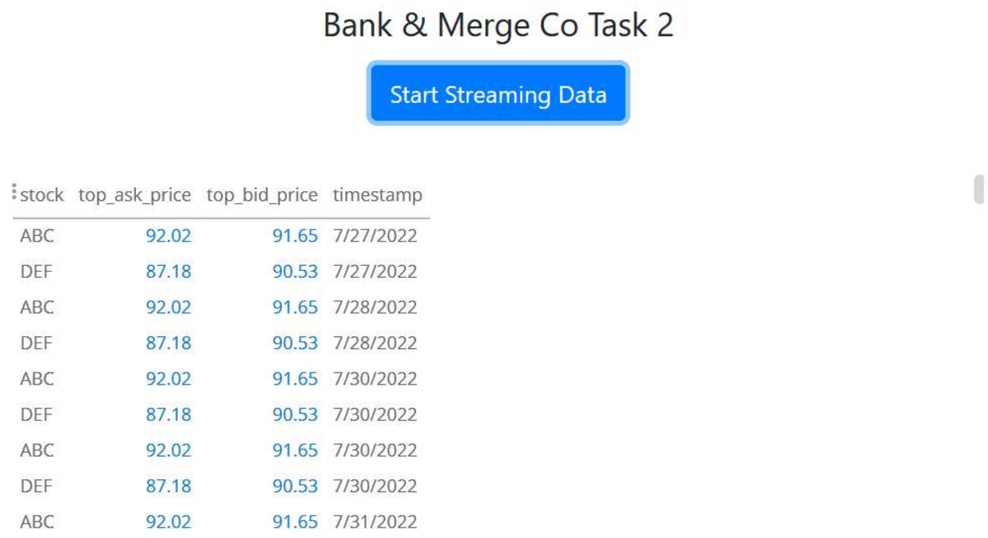
```
import React, { Component } from 'react';
import DataStreamer, { ServerRespond } from './DataStreamer';
import Graph from './Graph';
import './App.css';

/**
 * State declaration for <App />
 */
interface IState {
  data: ServerRespond[],
  showGraph: Boolean,
}

/**
 * The parent element of the react app.
 * It renders title, button and Graph react element.
 */
class App extends Component<{}, IState> {
  constructor(props: {}) {
    super(props);

    this.state = {
      // data saves the server responds.
      // We use this state to parse data down to the child element (Graph) as element property
      data: [],
    };
  }

```


server3.py M
DataStreamer.ts M
App.tsx M
src > App.tsx > App > getDataFromServer > interval > setInterval() callback > DataStreamer.getData() callback

```
    return (<Graph data={this.state.data}/>)
    }
  }

  /**
   * Get new data from server and update the state with the new data
   */
  getDataFromServer() {
    let x =0;
    const interval = setInterval(() => {
    DataStreamer.getData((serverResponds: ServerRespond[]) => {
      this.setState ({
        data: serverResponds,
        showGraph: true,
      });
    });
    x++;
    if(x>1000)
    {
      clearInterval(interval);
    }
      // Update the state by creating a new array of data that consists of
      // Previous data in the state and the new data from server
  }, 100);
  }
```

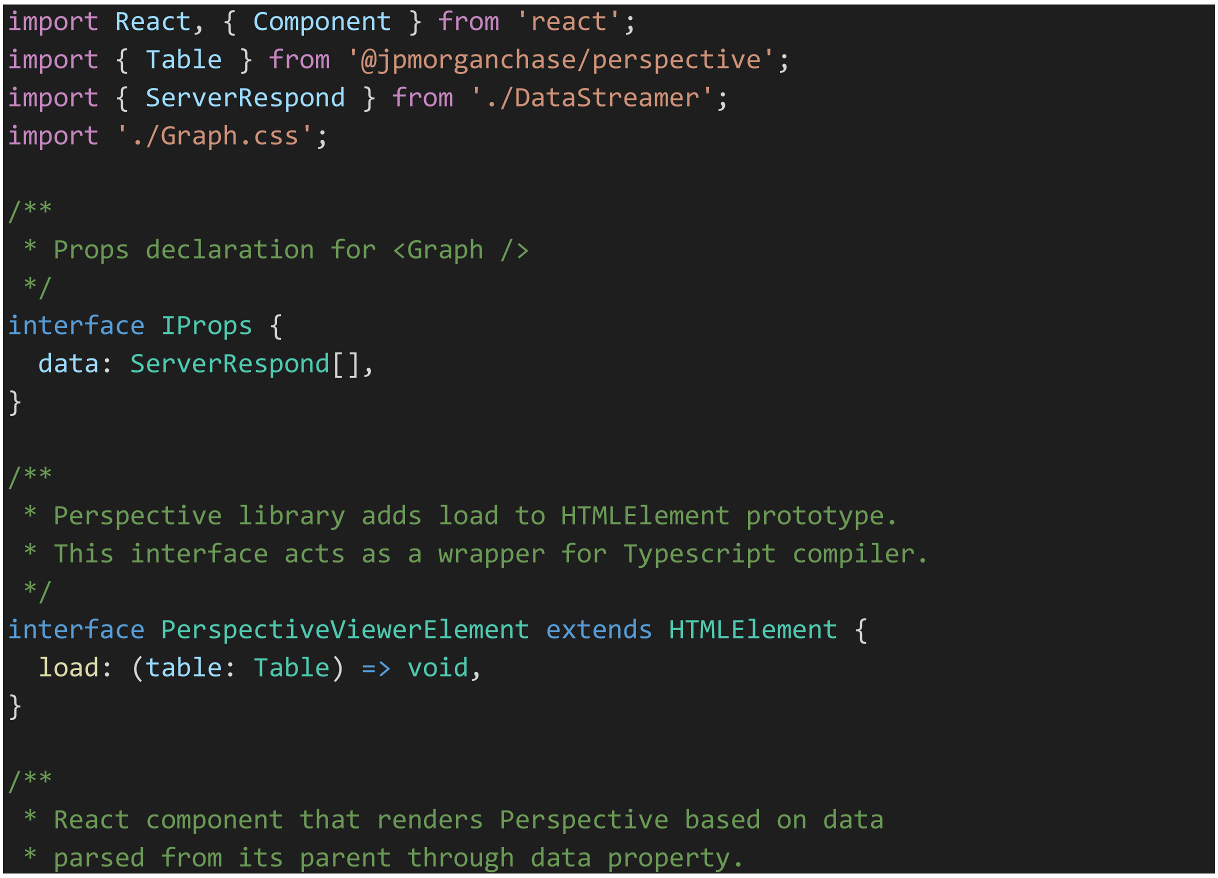


```
render() {
  return React.createElement('perspective-viewer');
}

componentDidMount() {
  // Get element to attach the table from the DOM.
  const elem = document.getElementsByTagName('perspective-viewer')[0] as unknown as PerspectiveViewerElement

  const schema = {
```

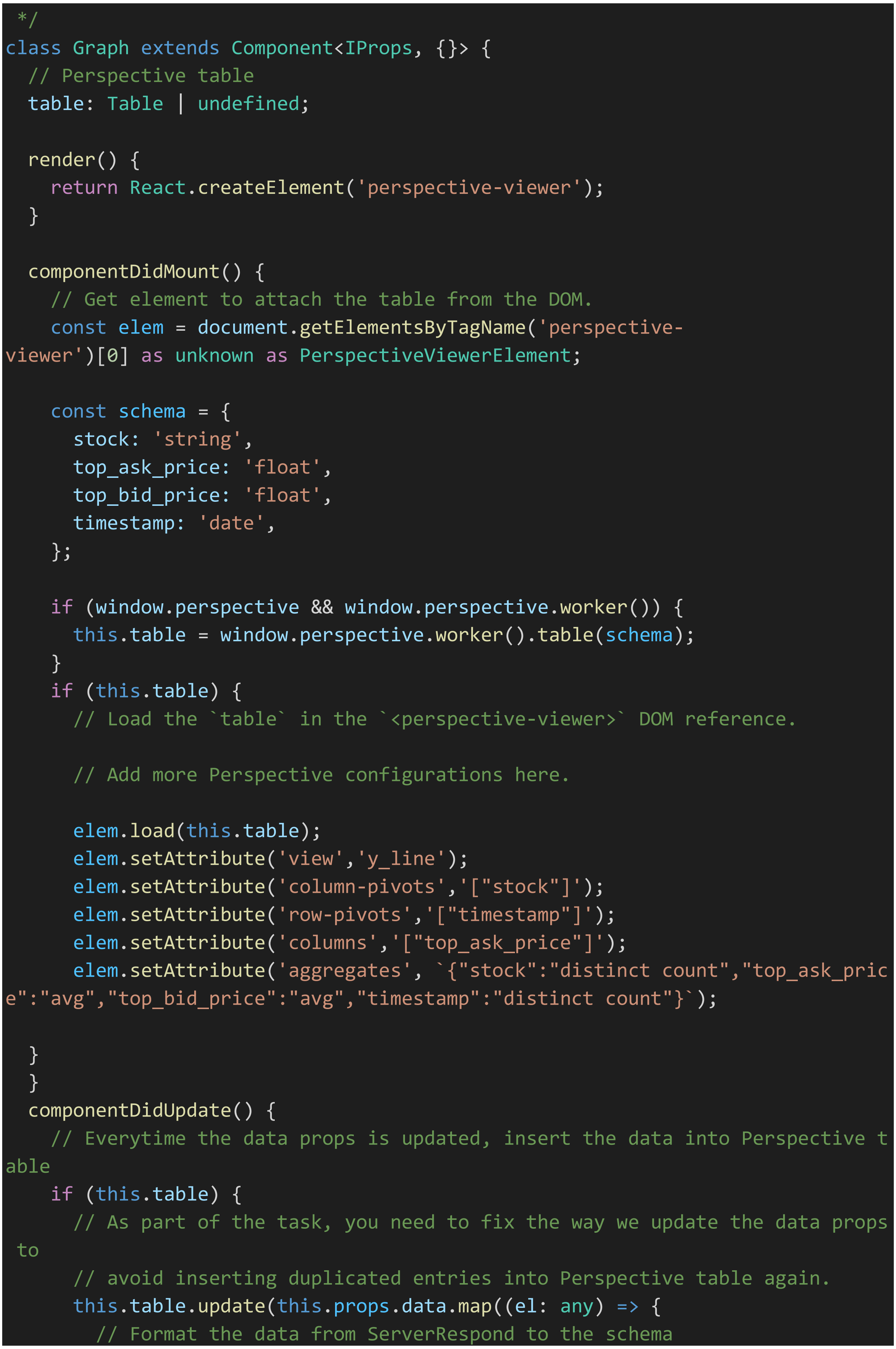


Bank & Merge Co Task 2

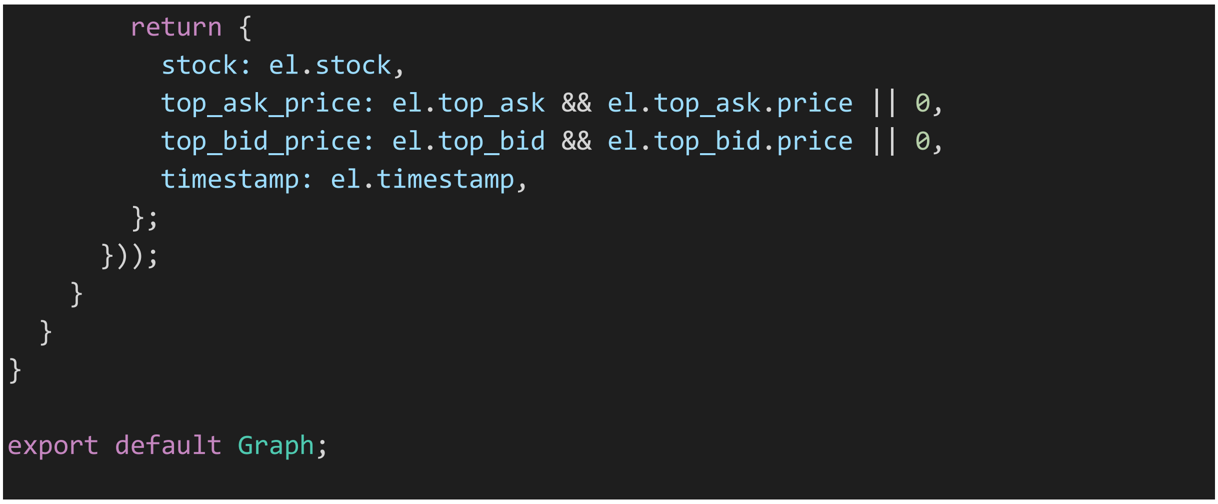
stock	top_ask_price	top_bid_price	timestamp
ABC	92.02	91.65	7/27/2022
DEF	87.18	90.53	7/27/2022
ABC	92.02	91.65	7/28/2022
DEF	87.18	90.53	7/28/2022
ABC	92.02	91.65	7/30/2022
DEF	87.18	90.53	7/30/2022
ABC	92.02	91.65	7/30/2022
DEF	87.18	90.53	7/30/2022
ABC	92.02	91.65	7/31/2022



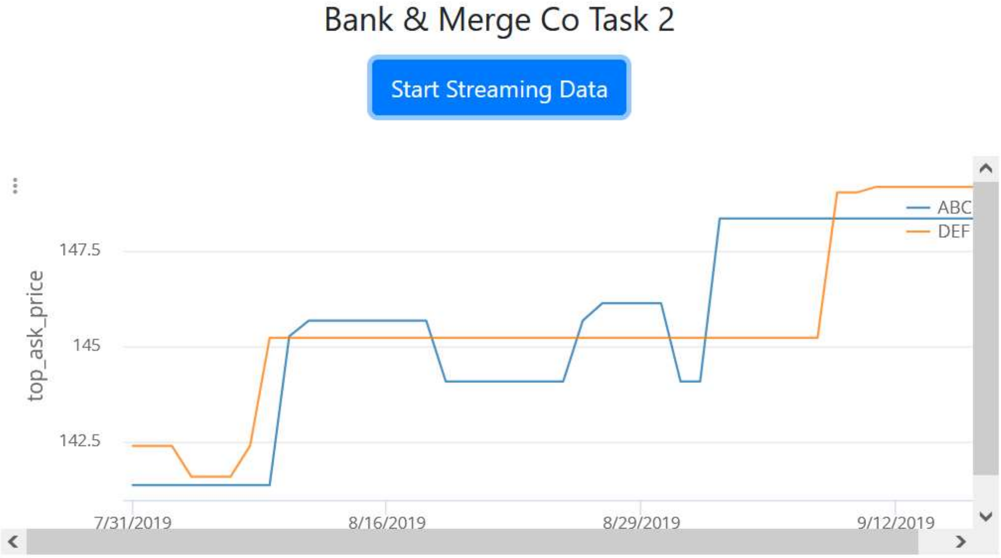
```
import React, { Component } from 'react';
import { Table } from '@jpmorganchase/perspective';
import { ServerRespond } from './DataStreamer';
import './Graph.css';

/**
 * Props declaration for <Graph />
 */
interface IProps {
  data: ServerRespond[],
}

/**
 * Perspective library adds load to HTMLElement prototype.
 * This interface acts as a wrapper for Typescript compiler.
 */
interface PerspectiveViewerElement extends HTMLElement {
  load: (table: Table) => void,
}

/**
 * React component that renders Perspective based on data
 * parsed from its parent through data property.
```



```
 */
class Graph extends Component<IProps, {}> {
  // Perspective table
  table: Table | undefined;

  render() {
    return React.createElement('perspective-viewer');
  }

  componentDidMount() {
    // Get element to attach the table from the DOM.
    const elem = document.getElementsByTagName('perspective-
viewer')[0] as unknown as PerspectiveViewerElement;

    const schema = {
      stock: 'string',
      top_ask_price: 'float',
      top_bid_price: 'float',
      timestamp: 'date',
    };

    if (window.perspective && window.perspective.worker()) {
      this.table = window.perspective.worker().table(schema);
    }
    if (this.table) {
      // Load the `table` in the `<perspective-viewer>` DOM reference.

      // Add more Perspective configurations here.

      elem.load(this.table);
      elem.setAttribute('view','y_line');
      elem.setAttribute('column-pivots','["stock"]');
      elem.setAttribute('row-pivots','["timestamp"]');
      elem.setAttribute('columns','["top_ask_price"]');
      elem.setAttribute('aggregates', `{"stock":"distinct count","top_ask_pric
e":"avg","top_bid_price":"avg","timestamp":"distinct count"}`);

  }
  }
  componentDidUpdate() {
    // Everytime the data props is updated, insert the data into Perspective t
able
    if (this.table) {
      // As part of the task, you need to fix the way we update the data props
 to
      // avoid inserting duplicated entries into Perspective table again.
      this.table.update(this.props.data.map((el: any) => {
        // Format the data from ServerRespond to the schema
```



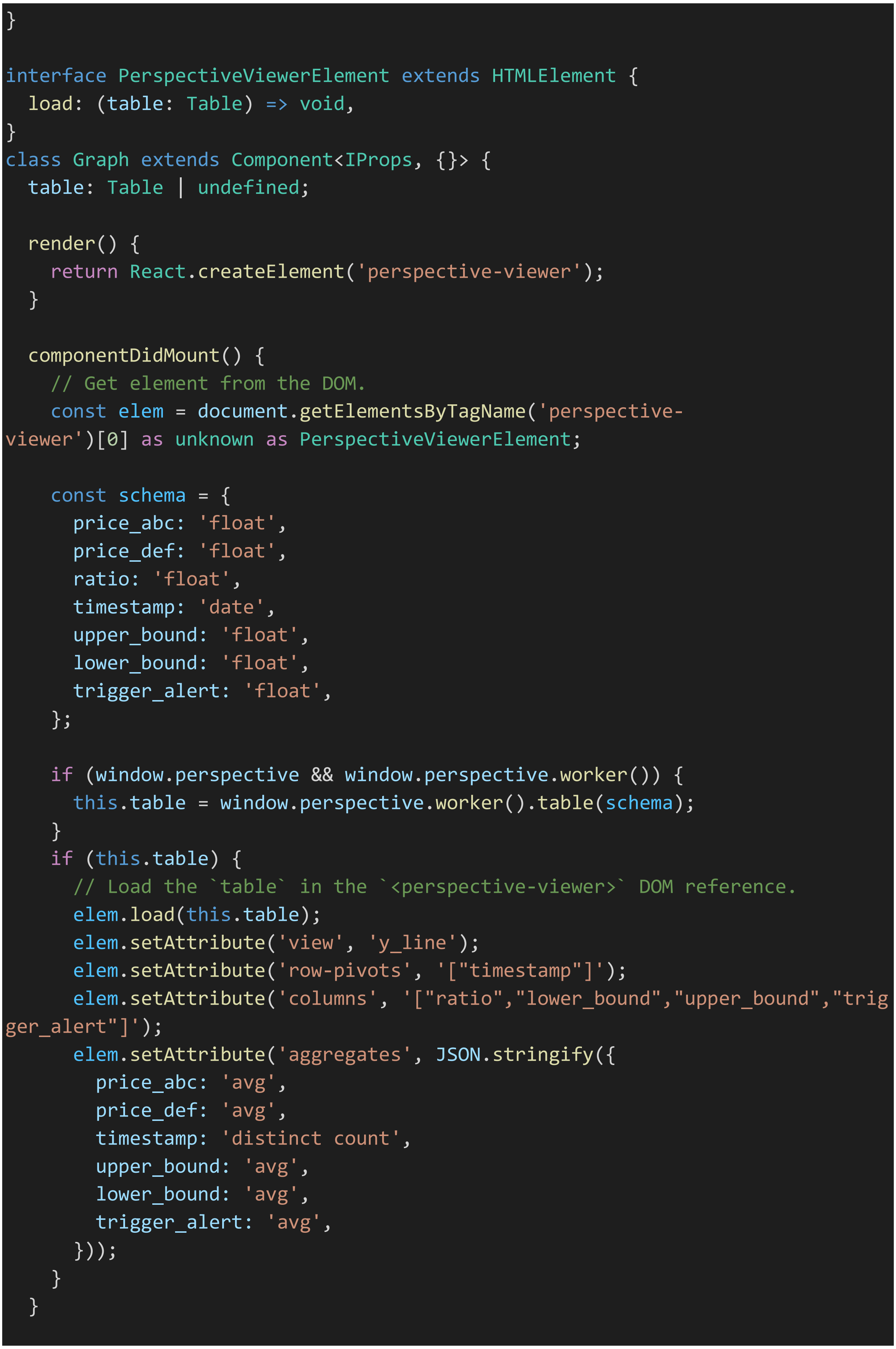
```
        return {
          stock: el.stock,
          top_ask_price: el.top_ask && el.top_ask.price || 0,
          top_bid_price: el.top_bid && el.top_bid.price || 0,
          timestamp: el.timestamp,
        };
      }));
    }
  }
}

export default Graph;
```



TASK 3

```
import { ServerRespond } from './DataStreamer';

export interface Row {
  price_abc: number,
  price_def: number,
  ratio: number,
  timestamp: Date,
  upper_bound: number,
  lower_bound: number,
  trigger_alert: number | undefined,
}


export class DataManipulator {
  static generateRow(serverResponds: ServerRespond[]): Row {
    const priceABC = (serverResponds[0].top_ask.price + serverResponds[0].top_bid.price)/2;
    const priceDEF = (serverResponds[1].top_ask.price + serverResponds[1].top_bid.price)/2;
    const ratio = priceABC / priceDEF;
    const upper_bound = 1 + 0.01;
    const lower_bound = 1 - 0.01;
    return {
      price_abc: priceABC,
      price_def: priceDEF,
      ratio,
      timestamp: serverResponds[0].timestamp > serverResponds[1].timestamp ?
                 serverResponds[0].timestamp : serverResponds[1].timestamp,
      upper_bound: upper_bound,
      lower_bound: lower_bound,
      trigger_alert: (ratio > upper_bound || ratio < lower_bound ) ? ratio : undefined,
      };
  }
}
```

```
import React, { Component } from 'react';
import { Table } from '@jpmorganchase/perspective';
import { ServerRespond } from './DataStreamer';
import { DataManipulator } from './DataManipulator';
import './Graph.css';

interface IProps {
  data: ServerRespond[],
```



```
}

interface PerspectiveViewerElement extends HTMLElement {
  load: (table: Table) => void,
}
class Graph extends Component<IProps, {}> {
  table: Table | undefined;

  render() {
    return React.createElement('perspective-viewer');
  }

  componentDidMount() {
    // Get element from the DOM.
    const elem = document.getElementsByTagName('perspective-viewer')[0] as unknown as PerspectiveViewerElement;

    const schema = {
      price_abc: 'float',
      price_def: 'float',
      ratio: 'float',
      timestamp: 'date',
      upper_bound: 'float',
      lower_bound: 'float',
      trigger_alert: 'float',
    };

    if (window.perspective && window.perspective.worker()) {
      this.table = window.perspective.worker().table(schema);
    }
    if (this.table) {
      // Load the `table` in the `<perspective-viewer>` DOM reference.
      elem.load(this.table);
      elem.setAttribute('view', 'y_line');
      elem.setAttribute('row-pivots', '["timestamp"]');
      elem.setAttribute('columns', '["ratio","lower_bound","upper_bound","trigger_alert"]');
      elem.setAttribute('aggregates', JSON.stringify({
        price_abc: 'avg',
        price_def: 'avg',
        timestamp: 'distinct count',
        upper_bound: 'avg',
        lower_bound: 'avg',
        trigger_alert: 'avg',
      }));
    }
  }
  componentDidUpdate() {
    if (this.table) {
      this.table.update([
        DataManipulator.generateRow(this.props.data),
      ]);
    }
  }
}

export default Graph;
```