Streaming Financial Data and Plots with Plotly

Yves Hilpisch
Plotcon, Oakland, 04. May 2017
SERVICES for financial institutions globally

EVENTS for Python quants & algorithmic traders

TRAINING about Python for finance & algorithmic trading

CERTIFICATION in cooperation with university

PLATFORM for browser-based data analytics

BOOKS about Python and finance

OPEN SOURCE Python library for financial analytics

http://tpq.io
Once logged in, you can then download the TWS application for your operating system. Starting the application then requires the previously chosen user name and password. TWS then might show up as in Trader Workstation after login with trial credentials on your desktop.

![Trader Workstation](https://pyalgo.tpq.io)

**Figure 58. Trader Workstation after login with trial credentials**

The arrangement of the different panels of TWS might be changed or new windows might pop up depending on what you request from the application. TWS break out window with option chain data shows a break out window with option chain data.
historical, unstructured data (news, tweets)  

historical, structured data (stock prices, P&L)  

real-time, unstructured data (news, tweets)  

real-time, structured data (stock ticks, exchange rates)  

real-time processing, real-time visualization
# Simple Tick Data Server with ZeroMQ

```python
import zmq
import time
import random

context = zmq.Context()
socket = context.socket(zmq.PUB)
socket.bind('tcp://0.0.0.0:5555')

AAPL = 100.

while True:
    AAPL += random.gauss(0, 1) * 0.5
    msg = 'AAPL %s %s' % (AAPL, time.strftime('%Y-%m-%d %H:%M:%S', time.localtime()))
    socket.send_string(msg)
```

# Simple Tick Data Client with ZeroMQ

```python
import zmq
import datetime

context = zmq.Context()
socket = context.socket(zmq.SUB)
socket.connect('tcp://0.0.0.0:5555')
socket.setsockopt_string(zmq.SUBSCRIBE, 'AAPL')

while True:
    msg = socket.recv_string()
    t = datetime.datetime.now()
    print('%s %s' % (t, msg))
```

Tick Server 

Tick Client 

Tick Data

```
<table>
<thead>
<tr>
<th>Stock</th>
<th>Price</th>
<th>Date</th>
<th>Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>AAPL</td>
<td>107.16</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>107.18</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>107.49</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>107.26</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>107.68</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>106.92</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>106.55</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>105.97</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>106.01</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>105.37</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>105.09</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>104.92</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>105.03</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>105.12</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>105.29</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>105.43</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
<tr>
<td>AAPL</td>
<td>105.60</td>
<td>2017-05-01</td>
<td>23:51:44</td>
</tr>
</tbody>
</table>
```
The Python Quants GmbH

Dr. Yves J. Hilpisch
+49 3212 112 9194
http://tpq.io | team@tpq.io
@dyjh