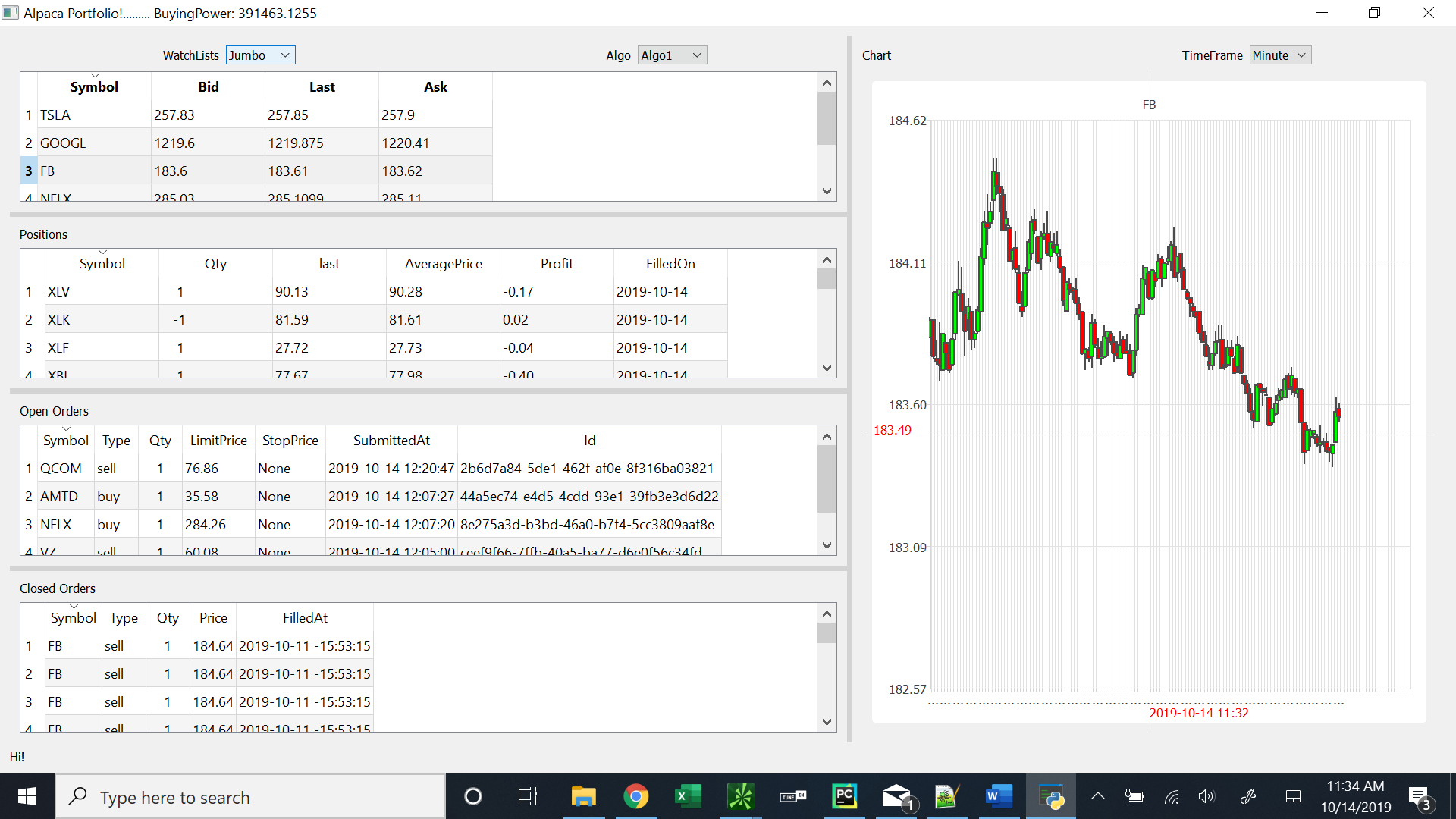
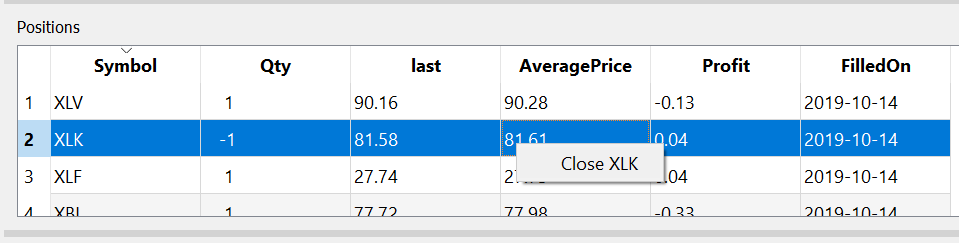
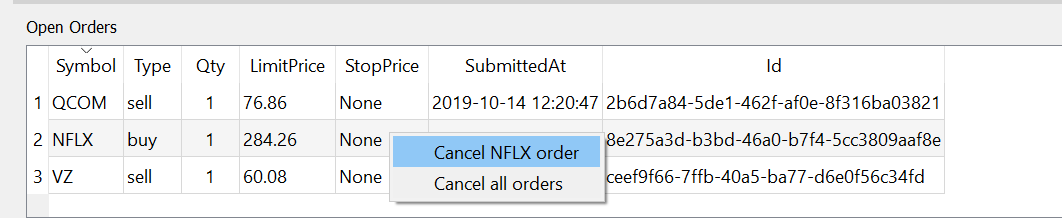
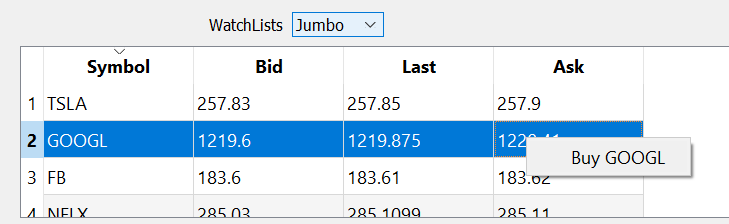
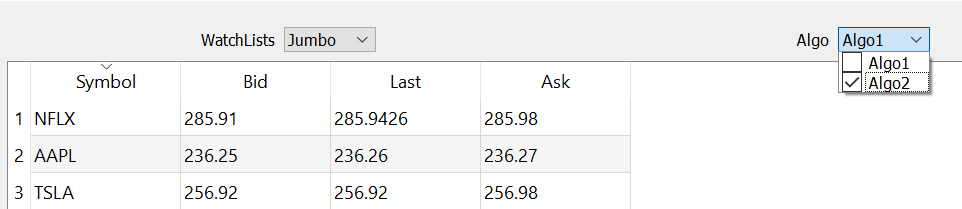
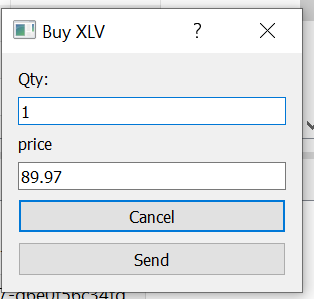
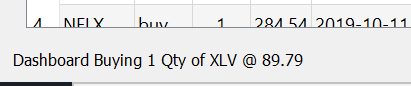
**Alpaca Dashboard**

****

**This GUI dashboard has the following features:**

* Shows all positions in your portfolio
  + Right click on a stock will allow you to close the position
  + Last price and profit are realtime data
  + 
* Shows all open orders in your portfolio
  + Right click on the stock will allow you cancel the selected stock or all open orders
  + 
* Shows all closed order for a selected stock
* Display real time candle stick chart for a selected stock
* Allows you to implement watchlists in your python code and display those in the dashboard
  + I have provided 4 watchlist, two of them are static watchlist and two are scanner watchlist which run at regular interval, you can add more with very few change to the code
  + Dashboard allows you to select the watchlist from a combo drop down
    - Bidprice and Askprice are realtime data from polygon streaming api
    - Right click on askprice allows you to buy the stock
    - Right click on bidprice allows you to sell the price
  + 
* Allows you to implement your own Algo routine to buy and sell in the python code and gives you the ability to select algo that should run in the background
  + 
* Clicking on the watchlist, positions, open order, closed orders stocks displays a candle stick chart with realtime data
  + Chart show a crosshair with the price and time
  + You can select the chart timeframe from the drop down, currently minute and day timeframe are implemented, you can add more with very few change to the code
  + Allows to zoom into the chart
  + Right click on the chart allows you buy or sell the stock at limit or stop price depending on where you click
  + Right click also allows you to reset zoom
  + 
* Buy and sell request opens a dialog, with selected price and qty as applicable
  + 
* The title shows the portfolio buying power
  + 
* Status line show all messages related to buy and sell, also show if the stock is traded by dashboard or from a algo



* **The Algo and Watchlist are samples, you can modify and implement your own**
* Reliability of realtime data depends on the Steaming API
  + All the symbols in the watchlist, positions and open order are subscribed
* GUI is using PySide2 library
  + You need to pip install PySide2
* The Python code uses Signal and Slots and I have kept the GUI and Data Logic totally independent, they talk to each other using the signal defined in the controller section of the code
* All of the code runs in its own thread

**There could be errors, but this is my first Jab at GUI in python, give it a try and modify and make it robust!.**