

Option B - Task 3: Data processing 2

Starting with the first task in which we have to build a candlestick chart, the following provides you a clear idea of how it looks like:



This data has been saved and made into a csv file called as “stock_data.csv” and then passed into a variable.

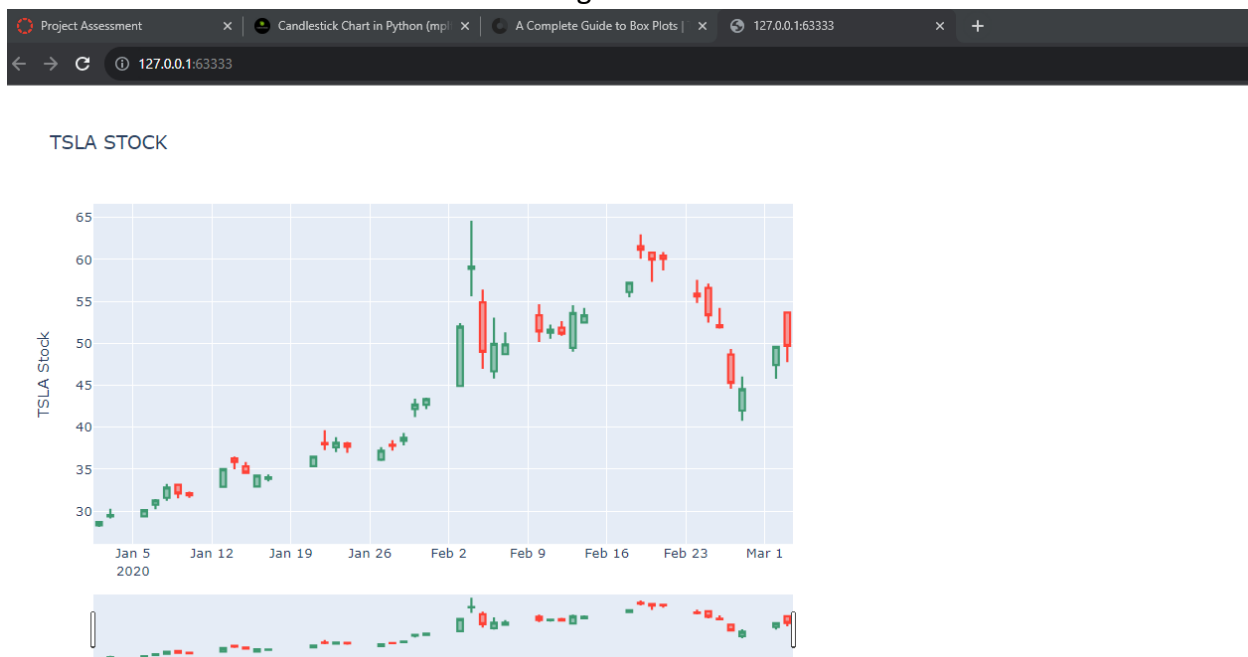
```
# creating a variable and storing the data
tsla_df = pd.read_csv('stock_data.csv', index_col= 0 ,parse_dates= True)
dt_range = pd.date_range(start = TRAIN_START, end = TRAIN_END)
tsla_df = tsla_df[tsla_df.index.isin(dt_range)]
tsla_df.head()
```

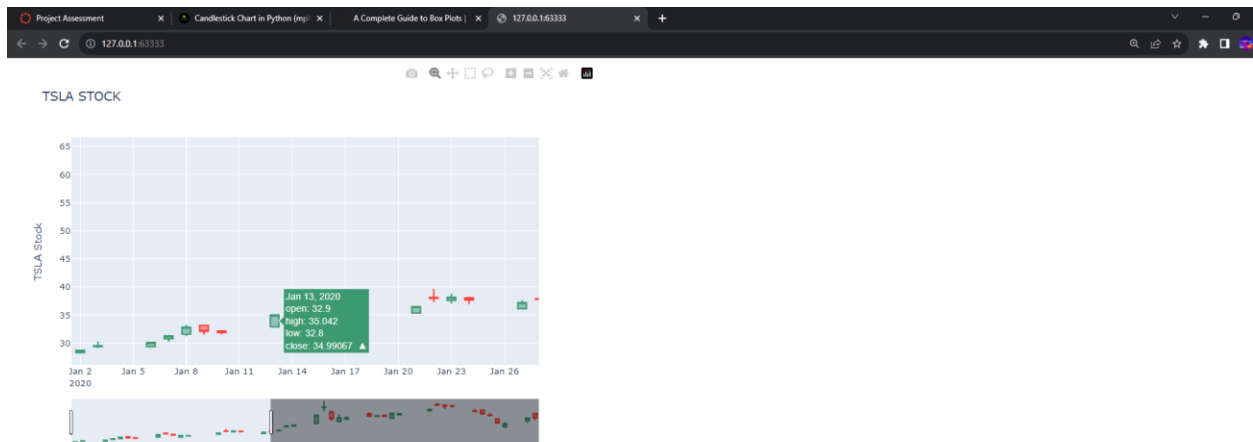
We then move onto using the function to make the candlestick Chart. I imported the mplfinance library to make this work after installing it from pip. There were different types of charts that I discovered like “Charles”, “mike”, “nightclouds” etc. which gives us different types of prebuilt custom charts.

```
# Making the Candle stick Chart
fplt.plot(
    tsla_df,
    type='candle',
    style='charles',
    title='TSLA, CandleStickChart',
    ylabel='Price ($)'
)
```

This function will give us a Candlestick Chart. Now some things to keep in mind is that I have used the start date and end date for only 2 months because feeding it more value not only consumes more time to train but also it makes the chart super messy. As I referred to the lecture present in the material given, we can also take the last 30 or maybe 50 values to show in the chart.

Moving on to the Second part of the requirement where we need to express the info in the individual candlesticks would look something like this:





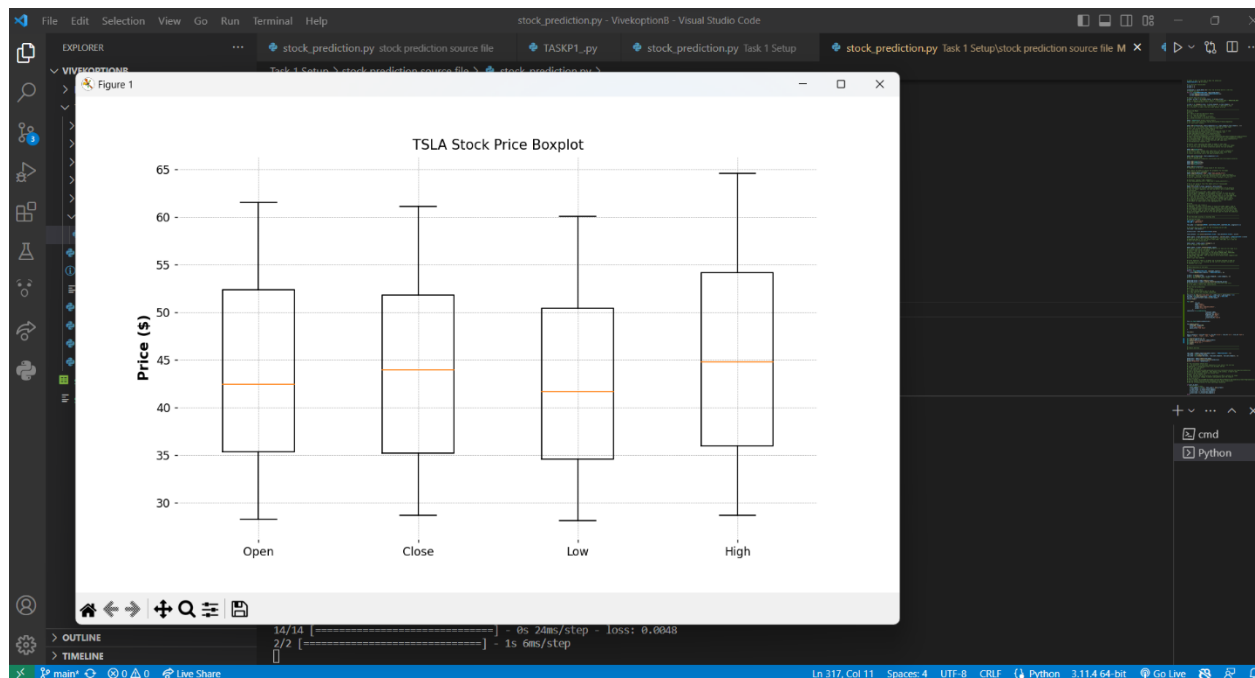
We can adjust the info with a slider that will only show the relevant info according to the number of n trading days that we want to see.

```
# Values for Individual Candlesticks
candlestick = go.Candlestick(
    x=tsla_df.index,
    open=tsla_df['Open'],
    high=tsla_df['High'],
    low=tsla_df['Low'],
    close=tsla_df['Close']
)
fig = go.Figure(data=[candlestick])
```

(variable) candlestick: Candlestick

This is the function which was referred from <https://coderzcolumn.com/tutorials/data-science/candlestick-chart-in-python-mplfinance-plotly-bokeh>

And at last we need to make a boxplot chart, I Used the same dataset I used for the other chart and referred to <https://www.educative.io/answers/how-to-make-a-boxplot-in-pandas> to make the function which results in something like this:



In conclusion, we used the certain libraries that come out from matplotlib and mplfinance , which helps us to visualize the data and enhanced how we look at the data That we have predicted.