

```
In [26]: !pip install pandas_datareader

Requirement already satisfied: pandas_datareader in /home/villacorta/anaconda3/lib/python3.7/site-packages (0.8.1)
Requirement already satisfied: pandas>=0.21 in /home/villacorta/anaconda3/lib/python3.7/site-packages (from pandas_datareader) (1.0.1)
Requirement already satisfied: requests>=2.3.0 in /home/villacorta/anaconda3/lib/python3.7/site-packages (from pandas_datareader) (2.22.0)
Requirement already satisfied: lxml in /home/villacorta/anaconda3/lib/python3.7/site-packages (from pandas_datareader) (4.5.0)
Requirement already satisfied: pytz>=2017.2 in /home/villacorta/anaconda3/lib/python3.7/site-packages (from pandas>=0.21->pandas_datareader) (2019.3)
Requirement already satisfied: python-dateutil>=2.6.1 in /home/villacorta/anaconda3/lib/python3.7/site-packages (from pandas>=0.21->pandas_datareader) (2.8.1)
Requirement already satisfied: numpy>=1.13.3 in /home/villacorta/anaconda3/lib/python3.7/site-packages (from pandas>=0.21->pandas_datareader) (1.18.1)
Requirement already satisfied: idna<2.9,>=2.5 in /home/villacorta/anaconda3/lib/python3.7/site-packages (from requests>=2.3.0->pandas_datareader) (2.8)
Requirement already satisfied: certifi>=2017.4.17 in /home/villacorta/anaconda3/lib/python3.7/site-packages (from requests>=2.3.0->pandas_datareader) (2019.11.28)
Requirement already satisfied: urllib3!=1.25.0,!1.25.1,<1.26,>=1.21.1 in /home/villacorta/anaconda3/lib/python3.7/site-packages (from requests>=2.3.0->pandas_datareader) (1.24.3)
Requirement already satisfied: chardet<3.1.0,>=3.0.2 in /home/villacorta/anaconda3/lib/python3.7/site-packages (from requests>=2.3.0->pandas_datareader) (3.0.4)
Requirement already satisfied: six>=1.5 in /home/villacorta/anaconda3/lib/python3.7/site-packages (from python-dateutil>=2.6.1->pandas>=0.21->pandas_datareader) (1.14.0)
```

```
In [2]: #nativo
from datetime import datetime

#terceros
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import pandas_datareader as web

/home/villacorta/anaconda3/lib/python3.7/site-packages/pandas_datareader/compat/__init__.py:7: FutureWarning: pandas.util.testing is deprecated. Use the functions in the public API at pandas.testing instead.
  from pandas.util.testing import assert_frame_equal
```

# ECONOMETRIA FINANCIERA

## EVALUACION SESION 1

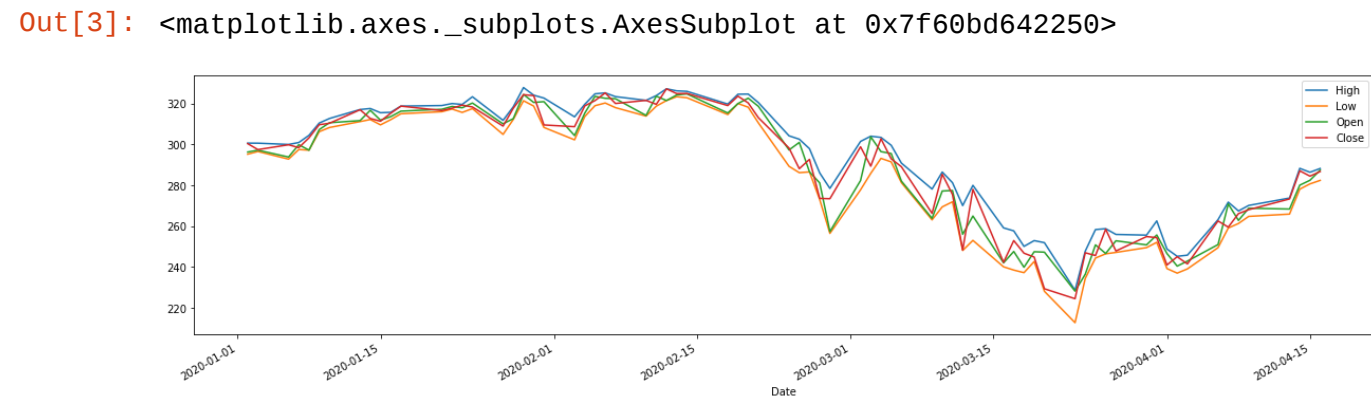
1.1- Descargar y graficar los precios Open, High, Low y Close de la acción (stock) de la acción de MICROSOFT, analice. Ayuda: El nombre financiero (ticker) de la acción de MICROSOFT es MSFT.

```
In [3]: start = datetime(2010, 1, 1)
end = datetime.now()

ticker = "AAPL"
cols_select = ['High', 'Low', 'Open', 'Close']

aapl = web.DataReader(ticker, 'yahoo', start, end)[cols_select]

# Grafico con respecto al año actual para mayor distinción
aapl['2020'].plot(figsize=(20, 5))
```



ANALISIS: Se aprecia una clara recuperacion que tiende a mantenerse constante, luego de una caída vertiginosa que inició en los primeros días de Marzo y que tuvo su pico mas bajo a fines de ese mismo mes, debido a la coyuntura del COVID-19.

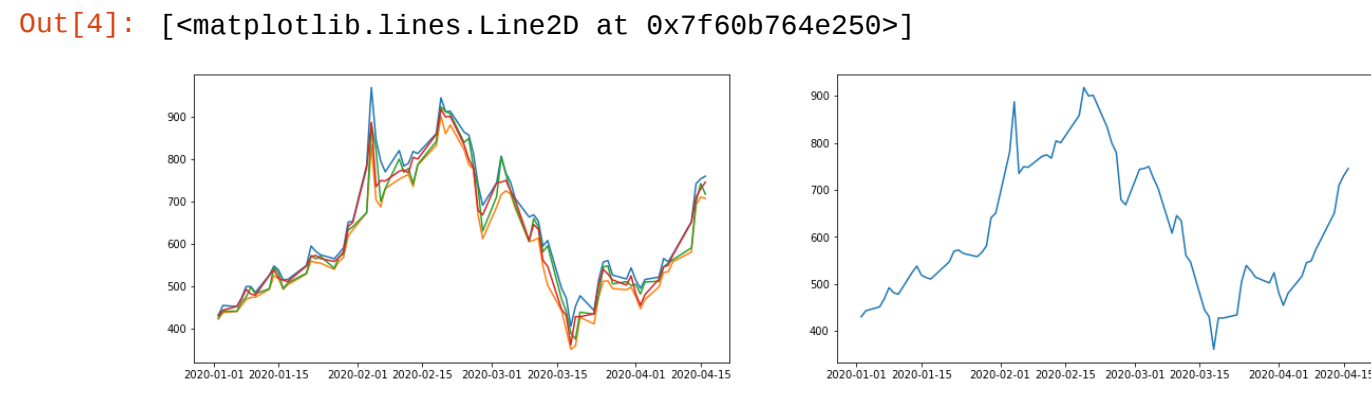
```
In [ ]:
```

1.2- Descargar y graficar los precios Open, High, Low y Close de la acción (stock) de la acción de TESLA, extraer y separar solo el precio Close, presentar en dos gráficos separados, a la izquierda los cuatro precios mencionados y a la derecha el precio Close, analice. Ayuda: El ticker de la acción de TESLA es TSLA.

```
In [4]: ticker_tesla = 'TSLA'

tsla = web.DataReader(ticker_tesla, 'yahoo', start, end)[cols_select]
only_close = tsla['Close'].copy()

# Grafico con respecto al año actual para mayor distinción
fig, axes = plt.subplots(nrows=1, ncols=2, figsize=(20, 5))
axes[0].plot(tsla['2020':])
axes[1].plot(only_close['2020':])
```



ANALISIS: Se aprecia una tendencia al alza, luego de una caída vertiginosa que inició a fines de Febrero y que tuvo su pico mas bajo en la quincena de Marzo, así como una recaída a inicios de Abril.

```
In [ ]:
```

1.3- Descargar la serie de precios de la acción de APLE y transformar la misma serie de precios Close de la acción de APLE en sus retornos, presentar la gráfica de la serie de precios arriba y la serie de los retornos abajo, analice.

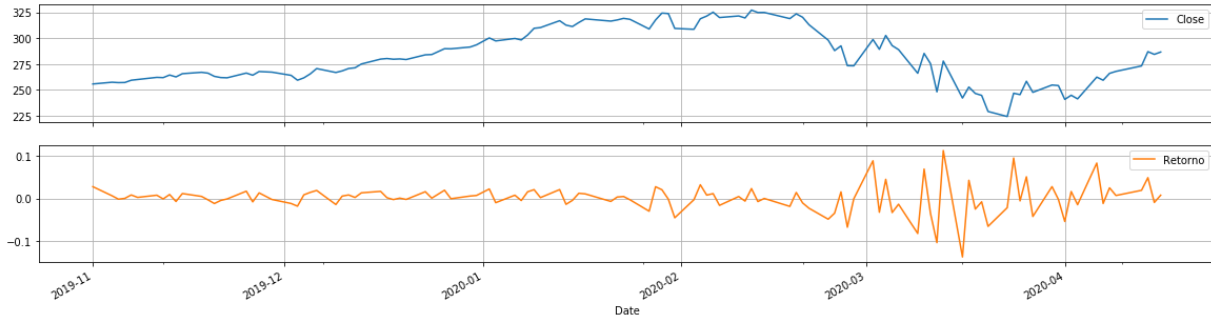
```
In [16]: ticker_apple = 'AAPL'

aapl = pd.DataFrame(web.DataReader(ticker_apple, 'yahoo', start, end)['Close'])

aapl['Retorno'] = np.log(aapl['Close']).diff()
#retorno = .dropna()

aapl['2019-11:'].plot(subplots=True, figsize=(20, 5), grid=True)
```

```
Out[16]: array([<matplotlib.axes._subplots.AxesSubplot object at 0x7f60b702e490>,
<matplotlib.axes._subplots.AxesSubplot object at 0x7f60b6c92550>],
dtype=object)
```



ANALISIS: Podemos ver que la gráfica de retorno no posee un comportamiento estacional, ya que su media no esta centrada en 0,

```
In [ ]:
```