In [1]:
```
import glob
import pandas as pd
from pandas import read_csv
import ipywidgets as widgets
from ipywidgets import interact, interact_manual
import plotly.express as px
import plotly.graph_objects as go
import os
from IPython.display import display, Image
import numpy as np
```

In [2]:
```
# get data file names
#path r'/Users/cindymendoncapaes/opt/anaconda3/lib/python3.8/site-packages/folium/finance project/Breakout US stocks'
path = r'C:\Users\baszo\Desktop\Breakout US stocks\A'
filenames = glob.glob(path + "/*.csv")

dfs = []
for filename in filenames:
    dfs.append(pd.read_csv(filename))

# Concatenate all data into one DataFrame
big_frame = pd.concat(dfs, ignore_index=True)
```

In [3]:
```
#set the column names
big_frame.columns = ['time', 'open', 'high', 'low', 'close', 'MA50', 'MA20', 'MA10', 'ADR', 'DV M', 'MA20 DV M', 'entry/exit']
big_frame
```

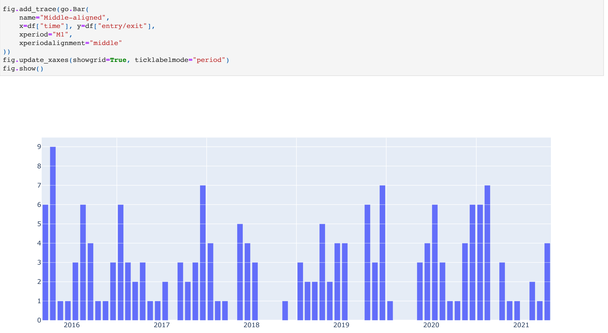
Out[3]:

	time	open	high	low	close	MA50	MA20	MA10	ADR	DV M	MA20 DV M	entry/exit
0	2014-04-16T15:30:00+02:00	1.32	1.3297	1.270	1.2900	1.526000	1.398500	1.40402	4.946169	0.124528	0.160346	NaN
1	2014-04-17T15:30:00+02:00	1.34	1.4400	1.320	1.3700	1.524400	1.393000	1.40302	5.228300	0.266812	0.167653	NaN
2	2014-04-21T15:30:00+02:00	1.37	1.3900	1.340	1.3637	1.523074	1.391185	1.38939	4.950582	0.202221	0.159332	NaN
3	2014-04-22T15:30:00+02:00	1.39	1.3900	1.350	1.3600	1.519874	1.391185	1.38638	4.762909	0.111773	0.143844	NaN
4	2014-04-23T15:30:00+02:00	1.33	1.4100	1.330	1.3900	1.512874	1.392185	1.37538	4.838097	0.110897	0.143069	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...
285934	2021-12-08T15:30:00+01:00	2013.76	2019.9650	1940.600	1988.8600	1803.385200	1880.377000	1875.55300	2.231629	497.250916	300.128051	NaN
285935	2021-12-09T15:30:00+01:00	1978.23	2008.2400	1951.860	1965.8300	1807.961800	1884.597500	1886.63100	2.283983	532.271101	308.351072	NaN
285936	2021-12-10T15:30:00+01:00	1975.79	2006.6700	1958.950	2003.0200	1814.062400	1890.404000	1905.51500	2.331454	319.769752	312.102683	NaN
285937	2021-12-13T15:30:00+01:00	2001.38	2001.3900	1963.630	1986.1000	1820.338800	1895.397500	1920.25600	2.362024	336.616320	317.529991	NaN
285938	2021-12-14T15:30:00+01:00	1982.98	2011.2100	1971.075	1986.5000	1827.218600	1899.130500	1937.19900	2.370320	87.654075	307.330713	NaN

285939 rows × 12 columns

In [11]:
```
#file=os.listdir(csv_directory));
#tickers = pd.read_csv(path+file, sep = ',')
combined_data = big_frame
#combined_data.columns = ['time', 'open', 'high', 'low', 'close', 'MA50', 'MA20', 'MA10', 'ADR', 'DV M', 'MA20 DV M', 'entry/exit']
#days_high = tickers["high"]
#days_low = tickers["low"]
#days_close = tickers["close"]
combined_data = combined_data[combined_data['entry/exit'].isin(["entry","exit"])]
combined_data['entry_price']= np.NaN
combined_data['exit_price'] = np.NaN
combined_data['profit'] = np.NaN
combined_data['risk'] = np.NaN
combined_data['R'] = np.NaN
combined_data.loc[combined_data['entry/exit'].isin(["entry"]), 'entry_price'] = (combined_data['low'] + combined_data['high']) /2
combined_data.loc[combined_data['entry/exit'].isin(["exit"]), 'exit_price'] = combined_data['close']
combined_data['entry_price'] = combined_data['entry_price'].fillna(method='ffill')
combined_data.loc[combined_data['entry/exit'].isin(["exit"]), 'profit'] = combined_data['exit_price']-combined_data['entry_price']
combined_data.loc[combined_data['entry/exit'].isin(["entry"]),'risk'] = combined_data['entry_price']-combined_data['low']
combined_data['risk'] = combined_data['risk'].fillna(method='ffill')
combined_data.loc[combined_data['entry/exit'].isin(["exit"]), 'R'] = combined_data['profit']/combined_data['risk']

from plotly import __version__
from plotly.offline import download_plotlyjs, init_notebook_mode, plot, iplot
import cufflinks as cf
cf.go_offline()

combined_data.drop(combined_data[combined_data['entry/exit'] == 'entry'].index, inplace=True)
combined_data.iplot(kind='scatter',x='ADR',y='R',mode='markers',size=10)
```

```
<ipython-input-11-66d8a0ee39ef>:9: SettingWithCopyWarning:


A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

<ipython-input-11-66d8a0ee39ef>:10: SettingWithCopyWarning:


A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

<ipython-input-11-66d8a0ee39ef>:11: SettingWithCopyWarning:


A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

<ipython-input-11-66d8a0ee39ef>:12: SettingWithCopyWarning:


A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

<ipython-input-11-66d8a0ee39ef>:13: SettingWithCopyWarning:


A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

C:\Users\baszo\anaconda3\lib\site-packages\pandas\core\indexing.py:966: SettingWithCopyWarning:


A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

<ipython-input-11-66d8a0ee39ef>:16: SettingWithCopyWarning:


A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy

<ipython-input-11-66d8a0ee39ef>:19: SettingWithCopyWarning:


A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```



Export to plot.ly »

In [4]:
```
#find the entries

big_frame = big_frame.loc[big_frame['entry/exit']=='entry']
big_frame = big_frame.replace('entry',1)
big_frame
```

Out[4]:

	time	open	high	low	close	MA50	MA20	MA10	ADR	DV M	MA20 DV M	entry/exit
1613	2020-09-11T15:30:00+02:00	0.684100	0.767000	0.674101	0.766300	0.652296	0.704835	0.697170	8.410072	0.813752	0.379531	1
3807	2021-08-02T15:30:00+02:00	12.890000	14.901500	12.890000	14.270000	10.992000	11.938000	12.738000	12.674249	2.589584	3.111956	1
5581	2017-11-13T15:30:00+01:00	1.140000	1.260000	1.140000	1.240000	0.850100	1.115210	1.144000	11.801697	2.615341	1.209093	1
5601	2017-12-12T15:30:00+01:00	1.600000	1.690000	1.540000	1.680000	1.257448	1.592000	1.632000	15.806241	0.985087	2.211793	1
7111	2016-08-23T15:30:00+02:00	13.820000	14.190000	13.740000	14.020000	11.892400	13.296500	13.724000	4.552783	2.458287	4.796923	1
...	...	...	...	...	...	...	...	...	...	...	...	...
272622	2019-07-15T15:30:00+02:00	7.060000	7.190000	7.050000	7.180000	6.489800	6.825000	7.002000	2.186272	5.381759	2.894671	1
274819	2016-04-07T15:30:00+02:00	3.190000	3.309900	3.170000	3.290000	2.743600	3.074500	3.065000	5.713740	44.724623	40.827642	1
279299	2021-04-15T15:30:00+02:00	188.060000	188.750000	187.100000	188.460000	182.615000	186.515500	186.840000	1.736751	180.065789	123.897420	1
281173	2016-03-16T14:30:00+01:00	35.852566	37.429836	35.644574	37.265176	34.680533	35.778035	36.589203	3.399712	24.023687	28.360884	1
284941	2017-12-28T15:30:00+01:00	709.500000	719.060000	709.120000	718.380000	648.010000	704.945000	706.046000	2.330987	198.668407	351.702049	1

179 rows × 12 columns

In [5]:
```
df = big_frame

fig = go.Figure()

fig.add_trace(go.Bar(
    name="Middle-aligned",
    x=df["time"], y=df["entry/exit"],
    xperiod="M1",
    xperiodalignment="middle"
))
fig.update_xaxes(showgrid=True, ticklabelmode="period")
fig.show()
```


In [12]:
```
csv_directory = 'C:/Users/baszo/Desktop/Breakout US stocks/A/'

@interact
def show_csv(file=os.listdir(csv_directory)):
    global code
    display(pd.read_csv(csv_directory+file, sep = ','))
    code = file.split('.csv')[0]

fdir = 'C:/Users/baszo/Desktop/Breakout US stocks/A/images/'

@interact
def show_images(ticker_name = code):
    list_code = os.listdir(fdir)
    list_match = [x for x in list_code if x.startswith(ticker_name + '_')]
    for file in list_match:
        display(Image(fdir+file))
```

```
interactive(children=(Dropdown(description='file', options=('BATS_AAU, 1D_e000d.csv', 'BATS_ACY, 1D_ff128.csv'…
interactive(children=(Text(value='BATS_AAU, 1D_e000d', description='ticker_name'), Output()), _dom_classes=('w…
```

In [13]:
```
path = 'C:/Users/baszo/Desktop/Breakout US stocks/A/'

@interact
def show_csv(file=os.listdir(path)):
    global tickers
    tickers = pd.read_csv(path+file, sep = ',')
    code = file.split('.csv')[0]


@interact
def display_time_series(param = tickers):
    list_code = os.listdir(path)
    fig = px.line(tickers, x=tickers['time'], y=param)
    fig.show()
```

```
interactive(children=(Dropdown(description='file', options=('BATS_AAU, 1D_e000d.csv', 'BATS_ACY, 1D_ff128.csv'…
C:\Users\baszo\anaconda3\lib\site-packages\pandas\core\ops\array_ops.py:253: FutureWarning:

elementwise comparison failed; returning scalar instead, but in the future will perform elementwise comparison

interactive(children=(Dropdown(description='param', options=('time', 'open', 'high', 'low', 'close', 'MA', 'MA…
```

In [ ]:
```

```