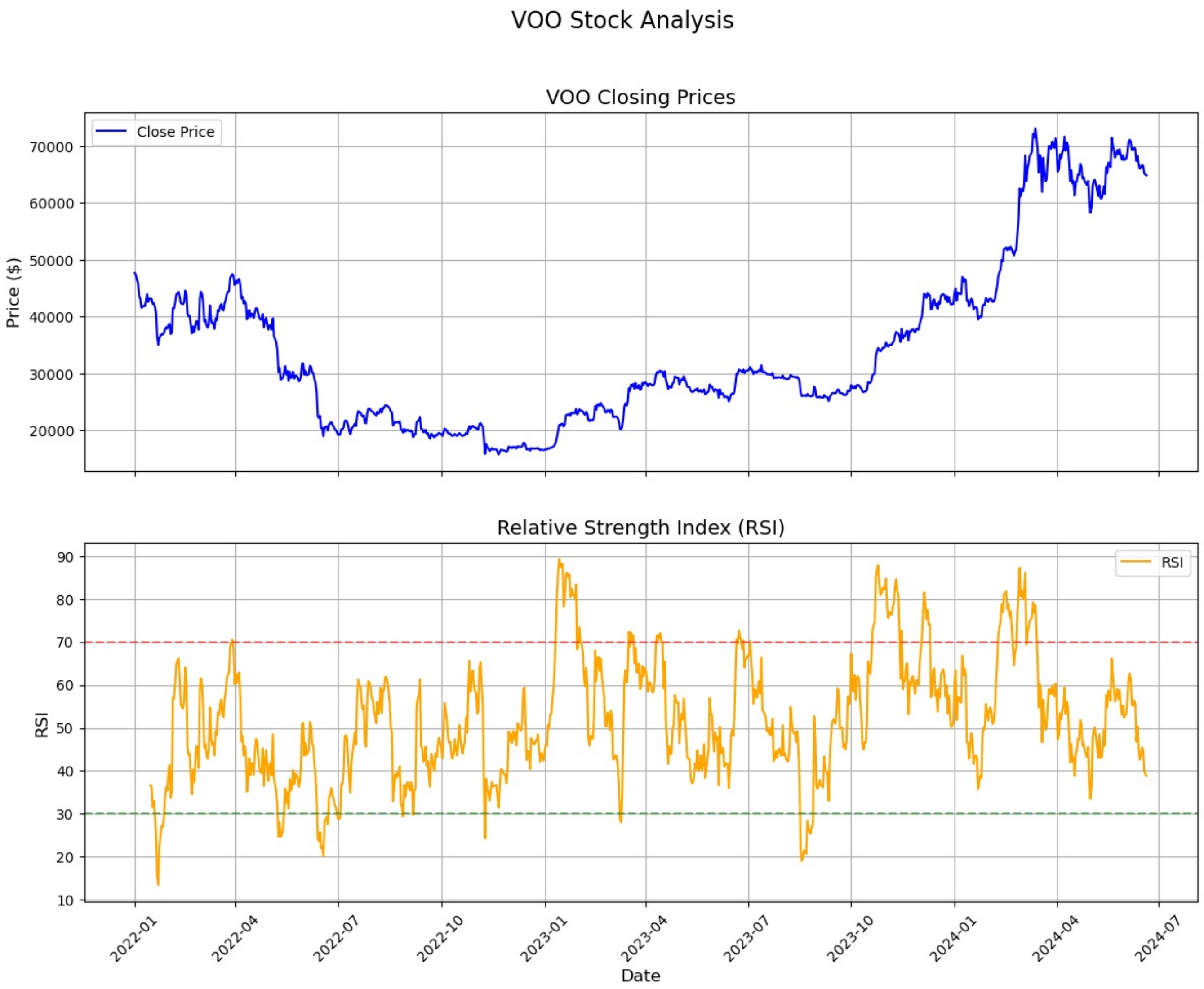
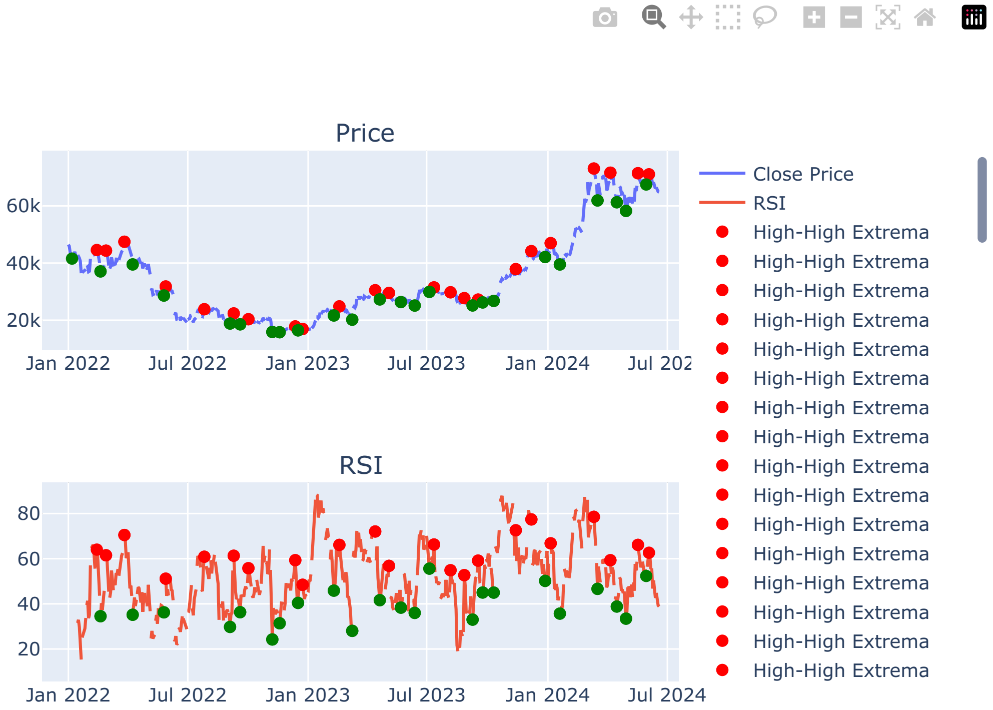
In [1]:

```python
import pandas as pd
import yfinance as yf
import matplotlib.pyplot as plt
from datetime import datetime
import pandas_ta as ta
import numpy as np
from scipy.signal import argrelextrema
from plotly.subplots import make_subplots
import plotly.graph_objects as go

# Define the ticker and date range
ticker = 'BTC-USD'
start_date = '2022-01-01'
end_date = datetime.today().strftime('%Y-%m-%d')

# Fetch the data
data = yf.download(ticker, start=start_date, end=end_date, interval='1d')

# Calculate RSI
data['RSI'] = ta.rsi(data['Close'], length=14)

# Plot the closing prices and RSI
fig, (ax1, ax2) = plt.subplots(2, 1, figsize=(14, 10), sharex=True)
fig.suptitle('VOO Stock Analysis', fontsize=16)

# Plot closing prices
ax1.plot(data.index, data['Close'], label='Close Price', color='blue')
ax1.set_title('VOO Closing Prices', fontsize=14)
ax1.set_ylabel('Price ($)', fontsize=12)
ax1.legend()
ax1.grid(True)

# Plot RSI
ax2.plot(data.index, data['RSI'], label='RSI', color='orange')
ax2.set_title('Relative Strength Index (RSI)', fontsize=14)
ax2.set_ylabel('RSI', fontsize=12)
ax2.axhline(70, linestyle='--', alpha=0.5, color='red')
ax2.axhline(30, linestyle='--', alpha=0.5, color='green')
ax2.legend()
ax2.grid(True)

# Rotate x-axis labels for better readability
plt.xticks(rotation=45)
plt.xlabel('Date', fontsize=12)
plt.show()

def find_extrema(data, order, how='hh'):
    extremas = None

    if how == 'hh':
        extremas = argrelextrema(data, comparator=np.greater, order=order)[0]
    if how == 'll':
        extremas = argrelextrema(data, comparator=np.less, order=order)[0]

    extremas_pairs = [extremas[i:i+2] for i in range(len(extremas) - 1)]
    return extremas_pairs

# Find extremas for price
hh_price = find_extrema(data['Close'].values, order=10, how='hh')
ll_price = find_extrema(data['Close'].values, order=10, how='ll')

fig = make_subplots(rows=2, cols=1, subplot_titles=(" Price", "RSI"))

fig.add_trace(go.Scatter(x=data.index, y=data['Close'], mode='lines', name='Close Price'), row=1, col=1)
fig.add_trace(go.Scatter(x=data.index, y=data['RSI'], mode='lines', name='RSI'), row=2, col=1)

for h_line in hh_price:
    fig.add_trace(go.Scatter(x=data.index[h_line], y=data['Close'][h_line], mode='markers', marker=dict(color='
for l_line in ll_price:
    fig.add_trace(go.Scatter(x=data.index[l_line], y=data['Close'][l_line], mode='markers', marker=dict(color='

for h_line_rsi in hh_price:
    fig.add_trace(go.Scatter(x=data.index[h_line_rsi], y=data['RSI'][h_line_rsi], mode='markers', marker=dict(c
for l_line_rsi in ll_price:
    fig.add_trace(go.Scatter(x=data.index[l_line_rsi], y=data['RSI'][l_line_rsi], mode='markers', marker=dict(c

fig.update_xaxes(rangebreaks=[dict(bounds=['sat', 'mon'])])
fig.show()
```

```
/Users/abdelkarimabdallah/anaconda3/lib/python3.11/site-packages/yfinance/base.py:48: FutureWarning: The defaul
t dtype for empty Series will be 'object' instead of 'float64' in a future version. Specify a dtype explicitly
to silence this warning.
  _empty_series = pd.Series()
[*********************100%%**********************]  1 of 1 completed
```


VOO Stock Analysis
VOO Closing Prices
Close Price
Price ($)
70000
60000
50000
40000
30000
20000
Relative Strength Index (RSI)
RSI
RSI
90
80
70
60
50
40
30
20
10
2022-01
2022-04
2022-07
2022-10
2023-01
2023-04
2023-07
2023-10
2024-01
2024-04
2024-07
Date


Price
60k
40k
20k
Jan 2022
Jul 2022
Jan 2023
Jul 2023
Jan 2024
Jul 202
RSI
80
60
40
20
Jan 2022
Jul 2022
Jan 2023
Jul 2023
Jan 2024
Jul 2024
Close Price
RSI
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema
High-High Extrema

```
In [ ]:
```