# /trading\_bot/main.py

import alpaca\_trade\_api as tradeapi

import numpy as np

import pandas as pd

import time

from datetime import datetime, timedelta

API\_KEY = 'your\_api\_key'

API\_SECRET = 'your\_api\_secret'

BASE\_URL = 'https://paper-api.alpaca.markets'

# Initialize Alpaca API

api = tradeapi.REST(API\_KEY, API\_SECRET, BASE\_URL, api\_version='v2')

def get\_historical\_data(symbol, start\_date, end\_date):

barset = api.get\_barset(symbol, 'day', start=start\_date, end=end\_date)

bars = barset[symbol]

data = [{

'time': bar.t.isoformat(),

'open': bar.o,

'high': bar.h,

'low': bar.l,

'close': bar.c,

'volume': bar.v

} for bar in bars]

return pd.DataFrame(data)

def calculate\_rsi(data, window=14):

delta = data['close'].diff()

gain = (delta.where(delta > 0, 0)).fillna(0)

loss = (-delta.where(delta < 0, 0)).fillna(0)

avg\_gain = gain.rolling(window=window, min\_periods=1).mean()

avg\_loss = loss.rolling(window=window, min\_periods=1).mean()

rs = avg\_gain / avg\_loss

rsi = 100 - (100 / (1 + rs))

return rsi

def moving\_average\_crossover(data, short\_window=40, long\_window=100):

data['short\_ma'] = data['close'].rolling(window=short\_window, min\_periods=1).mean()

data['long\_ma'] = data['close'].rolling(window=long\_window, min\_periods=1).mean()

data['signal'] = 0

data['signal'][short\_window:] = np.where(data['short\_ma'][short\_window:] > data['long\_ma'][short\_window:], 1, 0)

data['position'] = data['signal'].diff()

return data

def place\_order(symbol, qty, side, order\_type='market', time\_in\_force='gtc'):

try:

api.submit\_order(

symbol=symbol,

qty=qty,

side=side,

type=order\_type,

time\_in\_force=time\_in\_force

)

print(f"{side.capitalize()} order of {qty} {symbol} shares placed.")

except Exception as e:

print(f"An error occurred: {e}")

def trading\_bot():

symbol = 'AAPL'

start\_date = (datetime.now() - timedelta(days=365)).strftime('%Y-%m-%d')

end\_date = datetime.now().strftime('%Y-%m-%d')

qty = 10 # Quantity of shares to trade

# Fetch historical data

data = get\_historical\_data(symbol, start\_date, end\_date)

# Calculate RSI

data['rsi'] = calculate\_rsi(data)

# Calculate Moving Average Crossover

data = moving\_average\_crossover(data)

# Trading logic

if data['position'].iloc[-1] == 1 and data['rsi'].iloc[-1] < 30:

place\_order(symbol, qty, 'buy')

elif data['position'].iloc[-1] == -1 and data['rsi'].iloc[-1] > 70:

place\_order(symbol, qty, 'sell')

if \_\_name\_\_ == "\_\_main\_\_":

while True:

trading\_bot()

time.sleep(3600) # Run every hour