IBKRCampus

IBKRWebinars

www.ibkrcampus.com/webinars
www.ibkrcampus.eu/webinars

February 12, 2024

IBridgePy

Algorithmic Trading with IBKR and Backtesting Using IBridgePy

Dr. Hui Liu
Founder & CEO, Creator
Running River Investment LLC &
IBridgePy

Exchange and Industry Sponsored Webinars are presented by unaffiliated third parties. IBKR is not responsible for the content of these presentations. You should review the contents of each presentation and make your own judgment as to whether the content is appropriate for you. IBKR does not provide recommendations or advice. This presentation is not an advertisement or solicitation for new customers. It is intended only as an educational presentation.

IBKRCampus

IBKRWebinars

www.ibkrcampus.com/webinars
www.ibkrcampus.eu/webinars

Disclosure:

Options involve risk and are not suitable for all investors. For information on the uses and risks of options, you can obtain a copy of the Options Clearing Corporation risk disclosure document titled Characteristics and Risks of Standardized Options by visiting ibkr.com/occ. Multiple leg strategies, including spreads, will incur multiple transaction costs.

Futures are not suitable for all investors. The amount you may lose may be greater than your initial investment. Before trading futures, please read the CFTC Risk Disclosure. For a copy visit your local Interactive Brokers Website.

Security futures involve a high degree of risk and are not suitable for all investors. The amount you may lose may be greater than your initial investment. Before trading security futures, please read the Security Futures Risk Disclosure Statement. For a visit copy your local Interactive Brokers Website.

There is a substantial risk of loss in foreign exchange trading. The settlement date of foreign exchange trades can vary due to time zone differences and bank holidays. When trading across foreign exchange markets, this may necessitate borrowing funds to settle foreign exchange trades. The interest rate on borrowed funds must be considered when computing the cost of trades across multiple markets.

The Order types available through Interactive Brokers Trader Workstation are designed to help you limit your loss and/or lock in a profit. Market conditions and other factors may affect execution. In general, orders guarantee a fill or guarantee a price, but not both. In extreme market conditions, an order may either be executed at a different price than anticipated or may not be filled in the marketplace.

Throughout the lesson, please keep in mind that the examples discussed are purely for technical demonstration purposes, and do not constitute trading advice. Also, it is important to remember that placing trades in a paper account is recommended before any live trading.

There is a substantial risk of loss in trading futures and options. Past performance is not indicative of future results.

Any stock, options or futures symbols displayed are for illustrative purposes only and are not intended to portray recommendations.

IRS Circular 230 Notice: These statements are provided for information purposes only, are not intended to constitute tax advice which may be relied upon to avoid penalties under any federal, state, local or other tax statutes or regulations, and do not resolve any tax issues in your favor.

Interactive Brokers LLC is a member of NYSE FINRA SIPC

Interactive Brokers Canada Inc - Is a member of the Canadian Investment Regulatory Organization (CIRO) and Member - Canadian Investor Protection Fund

Interactive Brokers (U.K.) Limited is authorised and regulated by the Financial Conduct Authority. FCA Register Entry Number 208159

Interactive Brokers Ireland Limited is regulated by the Central Bank of Ireland. CBI, reference number C423427

Interactive Brokers Australia Pty. Ltd. - ABN 98 166 929 568 is licensed and regulated by the Australian Securities and Investments Commission AFSL: 453554

Interactive Brokers Singapore Pte. Ltd. - Is licensed and regulated by the Monetary Authority of Singapore. MAS, Licence No. CMS100917

Interactive Brokers Hong Kong Limited is regulated by the Hong Kong Securities and Futures Commission, and is a member of the SEHK and the HKFE

Interactive Brokers India Pvt. Ltd. member: NSE, BSE, SEBI. Regn. No. SEBI Registration No. INZ000217730; NSDL: IN-DP-NSDL-301-2008

Interactive Brokers Central Europe Zrt. is licensed and regulated by the Central Bank of Hungary (Magyar Nemzeti Bank) under no. H-EN-III-623/2020

As with all investments, your capital is at risk

Algorithmic Trading in Python and Backtesting Using IBridgePy

Dr. Hui Liu
https://www.linkedin.com/in/hui-liu-18356a14/
San Jose, CA, United States

IBKR webinar on Feb 12th 2024

Rent-a-Coder helps to build trading robots

www.IBridgePy.com

Contents

- Introduction about algorithmic trading and IBridgePy
- Get historical data
- Place a market order
- Build a simple buy-low-sell-high strategy
- Backtest using IBKR data
- Manage multiple IBKR accounts using IBridgePy

Algorithmic trading

Algorithmic trading is a method of executing orders using automated pre-programmed trading instructions, accounting for variables such as time, price, and volume, to send the orders out to the market over time.

Benefits of algo trading

- Less pressure
- Less human errors
- More free time
- Potential gains

Rent-a-Coder helps to build trading robots

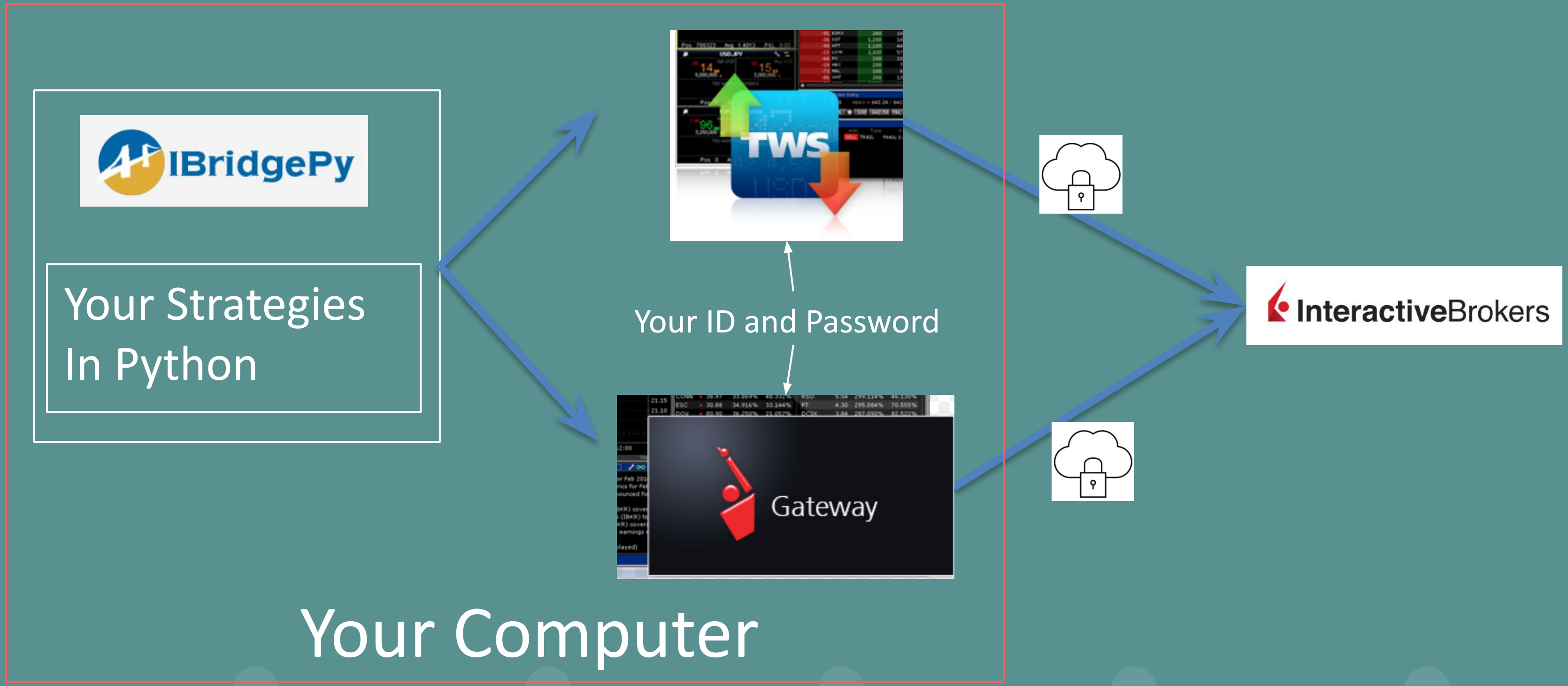
www.IBridgePy.com

Python choices

	Pros(advantages)	Cons(disadvantages)
Web-based trading platforms	● Huge data ● No installation ● Backtest	● NO PRIVACY ● Coding limitations ● Limited trading products
Low-level python wrappers	● Live trade ● Basic functions	● Hard to learn ● No backtest ● No support
Backtesting engines	● Backtest features ● Great community	● No live trading ● Limited trading products
Native API from brokers	● Live trade ● Some supports	● Hard to learn ● No backtest
Self-hosted solutions (IBridgePy)	● PRIVACY ● Easy to use ● Backtest and live	● Backtest needs data from providers ● a dedicated computer

Rent-a-Coder helps to build trading robots **www.IBridgePy.com**

Privacy protected



Your Strategies In Python

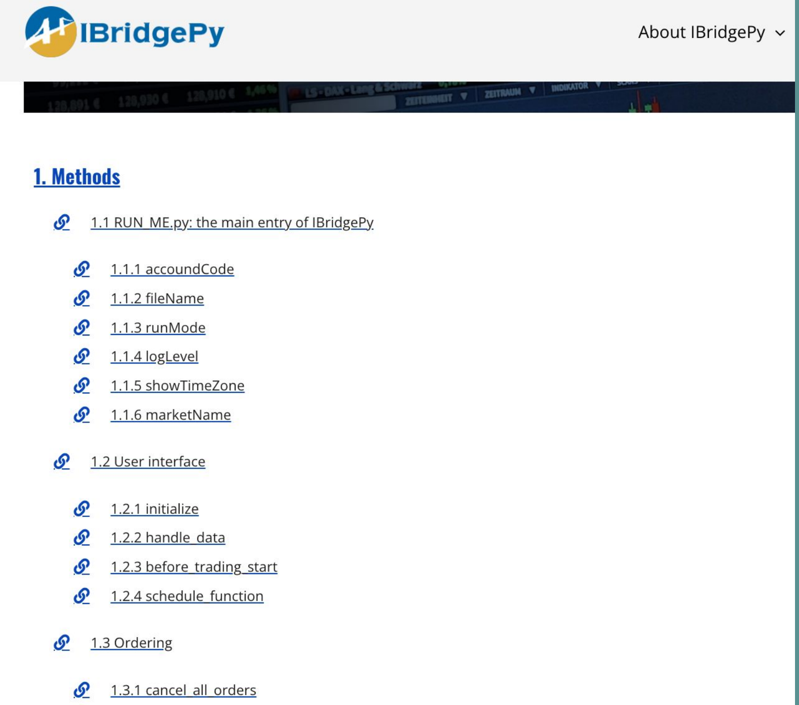
Your ID and Password

Your Computer

IBridgePy cornerstones

- Get historical data
- Retrieve account information
- Place orders

Learn more from https://ibridgepy.com/documentation/



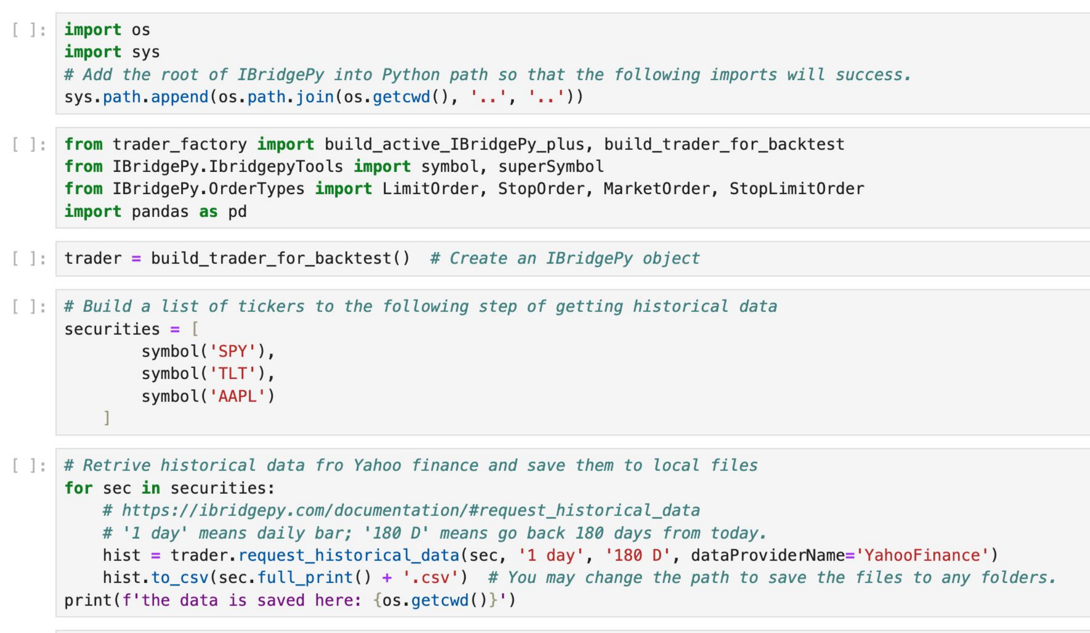
1. Methods

- 1.1 RUN_ME.py: the main entry of IBridgePy
  - 1.1.1 accoundCode
  - 1.1.2 fileName
  - 1.1.3 runMode
  - 1.1.4 logLevel
  - 1.1.5 showTimeZone
  - 1.1.6 marketName
- 1.2 User interface
  - 1.2.1 initialize
  - 1.2.2 handle_data
  - 1.2.3 before_trading_start
  - 1.2.4 schedule_function
- 1.3 Ordering
  - 1.3.1 cancel_all_orders

Rent-a-Coder helps to build trading robots www.IBridgePy.com

Get historical data

From Yahoo finance get_hist_from_yahoo_finance.ipynb



```
import os
import sys
# Add the root of IBridgePy into Python path so that the following imports will success.
sys.path.append(os.path.join(os.getcwd(), '..', '..'))
```

```
from trader_factory import build_active_IBridgePy_plus, build_trader_for_backtest
from IBridgePy.IbridgepyTools import symbol, superSymbol
from IBridgePy.OrderTypes import LimitOrder, StopOrder, MarketOrder, StopLimitOrder
import pandas as pd
```

```
trader = build_trader_for_backtest()  # Create an IBridgePy object
```

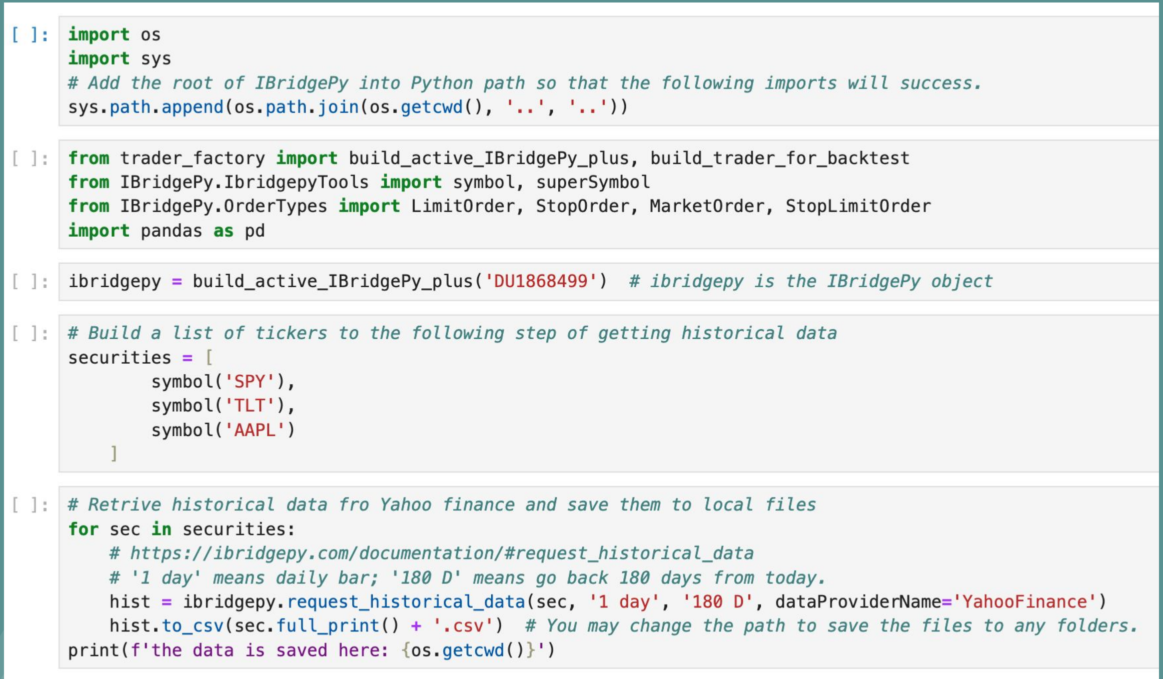
```
# Build a list of tickers to the following step of getting historical data
securities = [
        symbol('SPY'),
        symbol('TLT'),
        symbol('AAPL')
    ]
```

```
# Retrive historical data fro Yahoo finance and save them to local files
for sec in securities:
    # https://ibridgepy.com/documentation/#request_historical_data
    # '1 day' means daily bar; '180 D' means go back 180 days from today.
    hist = trader.request_historical_data(sec, '1 day', '180 D', dataProviderName='YahooFinance')
    hist.to_csv(sec.full_print() + '.csv')  # You may change the path to save the files to any folders.
print(f'the data is saved here: {os.getcwd()}')
```

Get historical data

From IBKR get_hist_from_IBKR.ipynb

- Run TWS or Gateway
- IBKR account
- Data package from IBKR

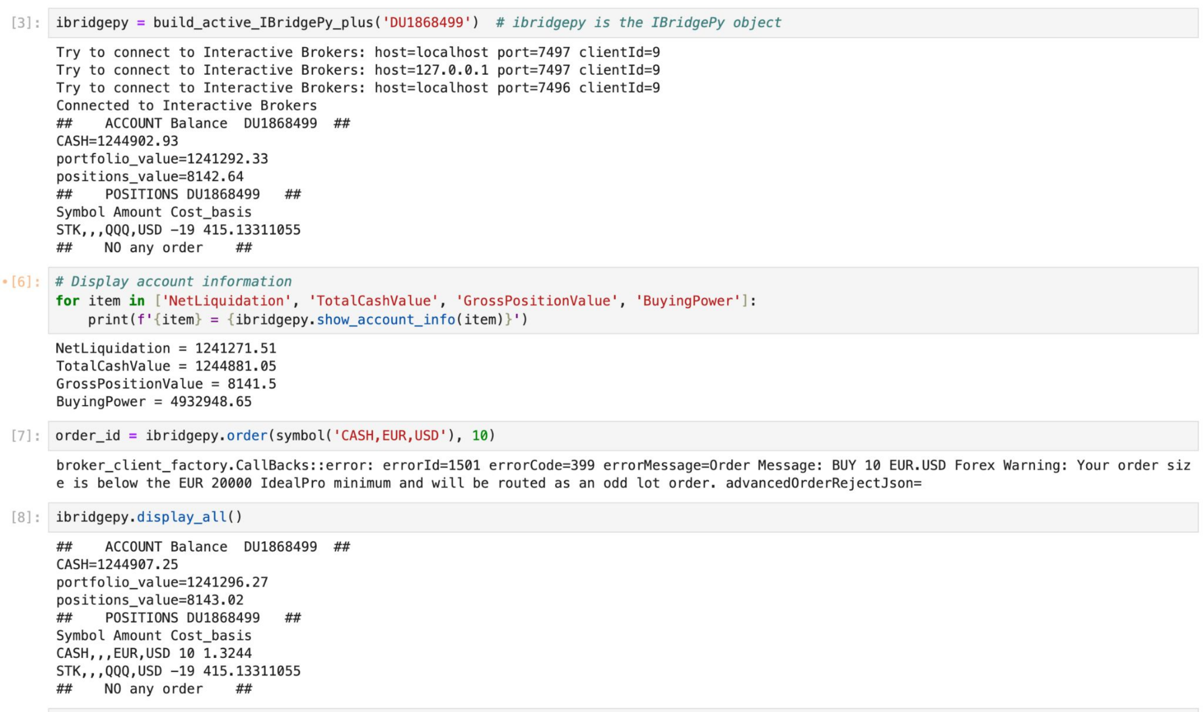


```
[ ]: import os
     import sys
     # Add the root of IBridgePy into Python path so that the following imports will success.
     sys.path.append(os.path.join(os.getcwd(), '..', '..'))

[ ]: from trader_factory import build_active_IBridgePy_plus, build_trader_for_backtest
     from IBridgePy.IbridgepyTools import symbol, superSymbol
     from IBridgePy.OrderTypes import LimitOrder, StopOrder, MarketOrder, StopLimitOrder
     import pandas as pd

[ ]: ibridgepy = build_active_IBridgePy_plus('DU1868499')  # ibridgepy is the IBridgePy object

[ ]: # Build a list of tickers to the following step of getting historical data
     securities = [
             symbol('SPY'),
             symbol('TLT'),
             symbol('AAPL')
         ]

[ ]: # Retrive historical data fro Yahoo finance and save them to local files
     for sec in securities:
         # https://ibridgepy.com/documentation/#request_historical_data
         # '1 day' means daily bar; '180 D' means go back 180 days from today.
         hist = ibridgepy.request_historical_data(sec, '1 day', '180 D', dataProviderName='YahooFinance')
         hist.to_csv(sec.full_print() + '.csv')  # You may change the path to save the files to any folders.
     print(f'the data is saved here: {os.getcwd()}')
```

Acc info and place order

From IBKR get_account_info_and_place_order.ipynb

- Run TWS or Gateway
- IBKR account
- Data package from IBKR

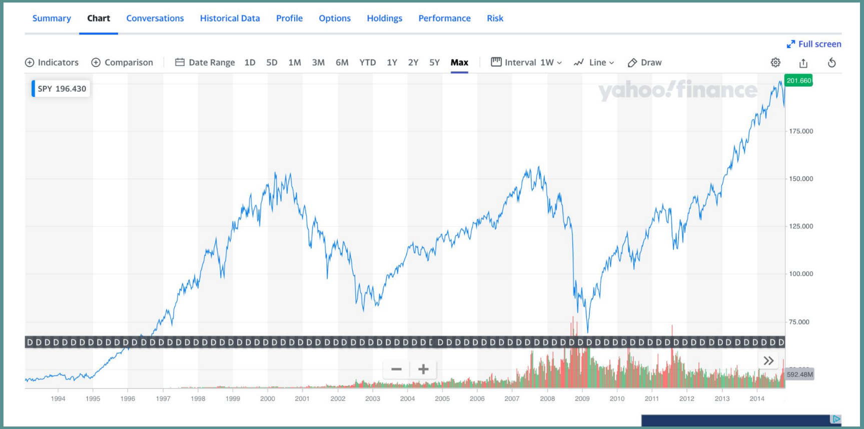


```
[3]: ibridgepy = build_active_IBridgePy_plus('DU1868499')  # ibridgepy is the IBridgePy object

     Try to connect to Interactive Brokers: host=localhost port=7497 clientId=9
     Try to connect to Interactive Brokers: host=127.0.0.1 port=7497 clientId=9
     Try to connect to Interactive Brokers: host=localhost port=7496 clientId=9
     Connected to Interactive Brokers
     ##    ACCOUNT Balance  DU1868499  ##
     CASH=1244902.93
     portfolio_value=1241292.33
     positions_value=8142.64
     ##    POSITIONS DU1868499   ##
     Symbol Amount Cost_basis
     STK,,,QQQ,USD -19 415.13311055
     ##    NO any order    ##

•[6]: # Display account information
     for item in ['NetLiquidation', 'TotalCashValue', 'GrossPositionValue', 'BuyingPower']:
         print(f'{item} = {ibridgepy.show_account_info(item)}')

     NetLiquidation = 1241271.51
     TotalCashValue = 1244881.05
     GrossPositionValue = 8141.5
     BuyingPower = 4932948.65

[7]: order_id = ibridgepy.order(symbol('CASH,EUR,USD'), 10)

     broker_client_factory.CallBacks::error: errorId=1501 errorCode=399 errorMessage=Order Message: BUY 10 EUR.USD Forex Warning: Your order siz
     e is below the EUR 20000 IdealPro minimum and will be routed as an odd lot order. advancedOrderRejectJson=

[8]: ibridgepy.display_all()

     ##    ACCOUNT Balance  DU1868499  ##
     CASH=1244907.25
     portfolio_value=1241296.27
     positions_value=8143.02
     ##    POSITIONS DU1868499   ##
     Symbol Amount Cost_basis
     CASH,,,EUR,USD 10 1.3244
     STK,,,QQQ,USD -19 415.13311055
     ##    NO any order    ##
```

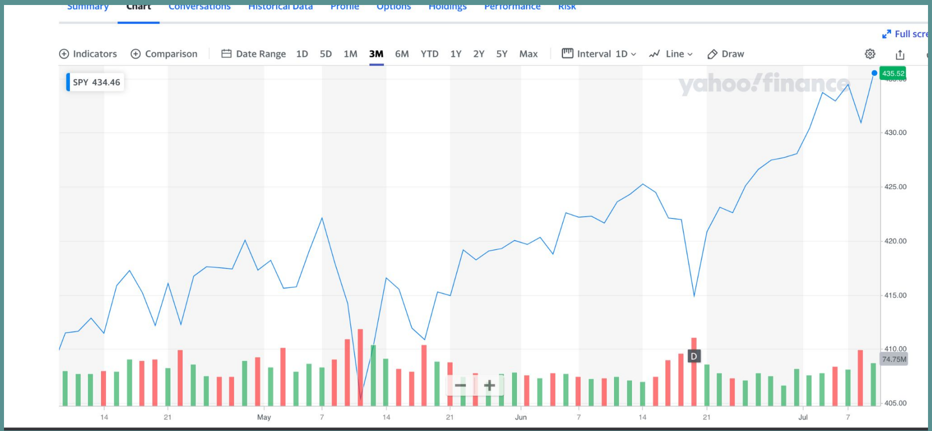
Rent-a-Coder helps to build trading robots

www.IBridgePy.com

Come up a trading idea



Good trend in long term



Swing in short term

My example strategy: If the price drops from yesterday, I place “Buy” order. Otherwise, don’t hold any positions. Do you think it is a profitable strategy?

Disclosure:
Any trading symbols, entities or investment products displayed are for illustrative purposes only and are not intended to portray recommendations. Past performance is not necessarily indicative of future results.

Build a machine learning model

Real time demo: buy_low_sell_high_model.ipynb

- Retrieve historical data
- Prepare data
- Build a machine learning model
- Visualize

Build a trading robot

```
def initialize(context):                                          # This function run only once at the beginning
    context.security = symbol(“SPY”)                              # Define a security, SP500 ETF
    schedule_function(buy_low_sell_high,                          # Schedule a time to run another function buy_low_sell_high
            date_rule=date_rules.every_day(),                     # The function of buy_low_sell_high is triggered every trading day
            time_rule=time_rules.market_close(minutes=1))         # at 15:59PM EST, one minute before the market close

def buy_low_sell_high(context, data):                             # The logic of buy low sell high are made in this function
    hist = request_historical_data(context.security, ”1 day”, “2 D”)  # Retrieve historical data, daily bars, go back 2 trading days
    close_yesterday = hist[“close”][-2]
    close_today = hist[“close”][-1]
    if close_today > close_yesterday:
        order_target_percent(context.security, 0.0)               # Sell off all SPY positions if any
    else:
        order_target_percent(context.security, 1.0)               # Buy SPY use 100% of portfolio
```

Rent-a-Coder helps to build trading robots **www.IBridgePy.com**

Backtest

- Use historical data directly from IBKR
- Use any 3rd party data providers
- Backtest minute-by-minute, even second-by-second
- Speed up!

More details on https://ibridgepy.com/tutorials/

Backtest

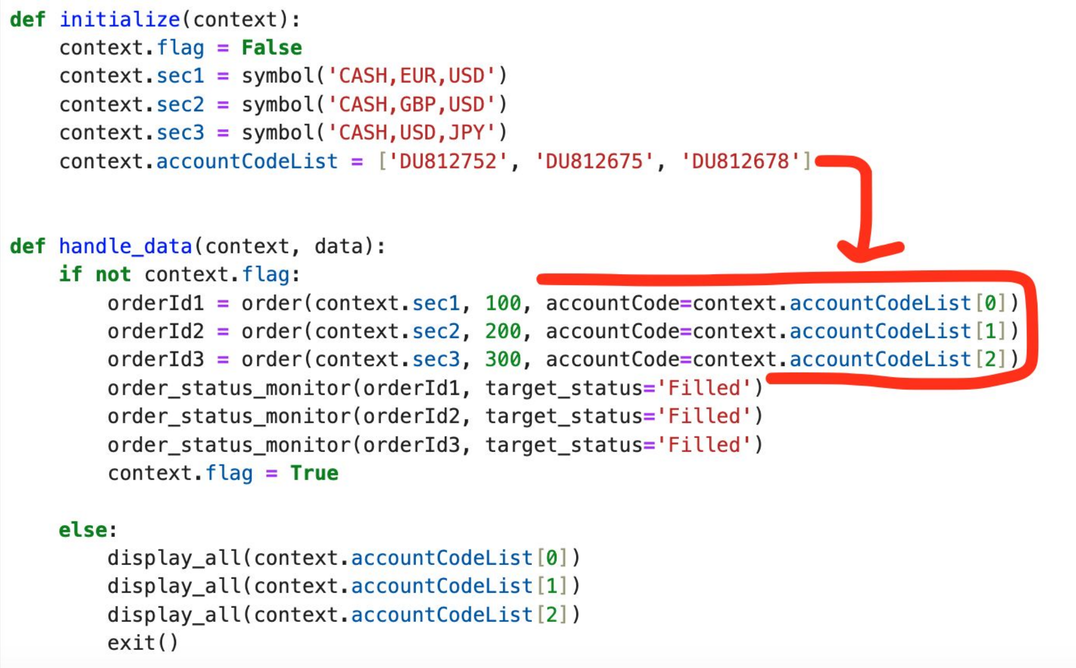
Real time demo: TEST_ME.ipynb

- Choose a strategy py file
- Pick a startTime and endTime for backtesting
- Confirm data provider

Live trade

- No code change at all !
- RUN_ME.py

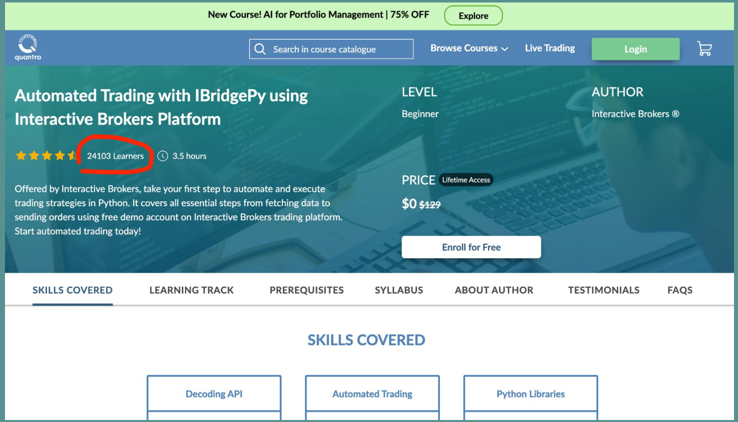
Manage multiple IBKR accounts



```python
def initialize(context):
    context.flag = False
    context.sec1 = symbol('CASH,EUR,USD')
    context.sec2 = symbol('CASH,GBP,USD')
    context.sec3 = symbol('CASH,USD,JPY')
    context.accountCodeList = ['DU812752', 'DU812675', 'DU812678']


def handle_data(context, data):
    if not context.flag:
        orderId1 = order(context.sec1, 100, accountCode=context.accountCodeList[0])
        orderId2 = order(context.sec2, 200, accountCode=context.accountCodeList[1])
        orderId3 = order(context.sec3, 300, accountCode=context.accountCodeList[2])
        order_status_monitor(orderId1, target_status='Filled')
        order_status_monitor(orderId2, target_status='Filled')
        order_status_monitor(orderId3, target_status='Filled')
        context.flag = True

    else:
        display_all(context.accountCodeList[0])
        display_all(context.accountCodeList[1])
        display_all(context.accountCodeList[2])
        exit()
```

Courses about Algo Trading

Free course about IBridgePy



Automated Trading with IBridgePy using Interactive Brokers Platform

★★★★½ 24103 Learners 3.5 hours

Offered by Interactive Brokers, take your first step to automate and execute trading strategies in Python. It covers all essential steps from fetching data to sending orders using free demo account on Interactive Brokers trading platform. Start automated trading today!

LEVEL
Beginner

AUTHOR
Interactive Brokers ®

PRICE Lifetime Access
$0 ~~$129~~

Enroll for Free

SKILLS COVERED

Decoding API

Automated Trading

Python Libraries

Special 22% discount for other paid course:
https://tinyurl.com/algocourses
Discount code: HUIWEB

Rent-a-Coder helps to build trading robots

www.IBridgePy.com

Thank you so much for your attention

Check out our well known Rent-a-Coder service if you need help on coding

IBridgePy@gmail.com

Dr. Hui Liu, from San Jose CA, U.S.A

www.IBridgePy.com

An easy-to-use Python platform to backtest and live trade