

HƯỚNG DẪN SỬ DỤNG FASTCONNECT TRADING

FastConnect Trading (FCTrading) là bộ API cung cấp hỗ trợ khách hàng đặt, sửa, hủy lệnh và truy vấn thông tin tài khoản. SSI hỗ trợ môi trường Python và Node JS/

1. Quy trình tích hợp

Quy trình tích hợp bao gồm đăng ký sử dụng dịch vụ, Key Credential

1.1. Đăng ký dịch vụ

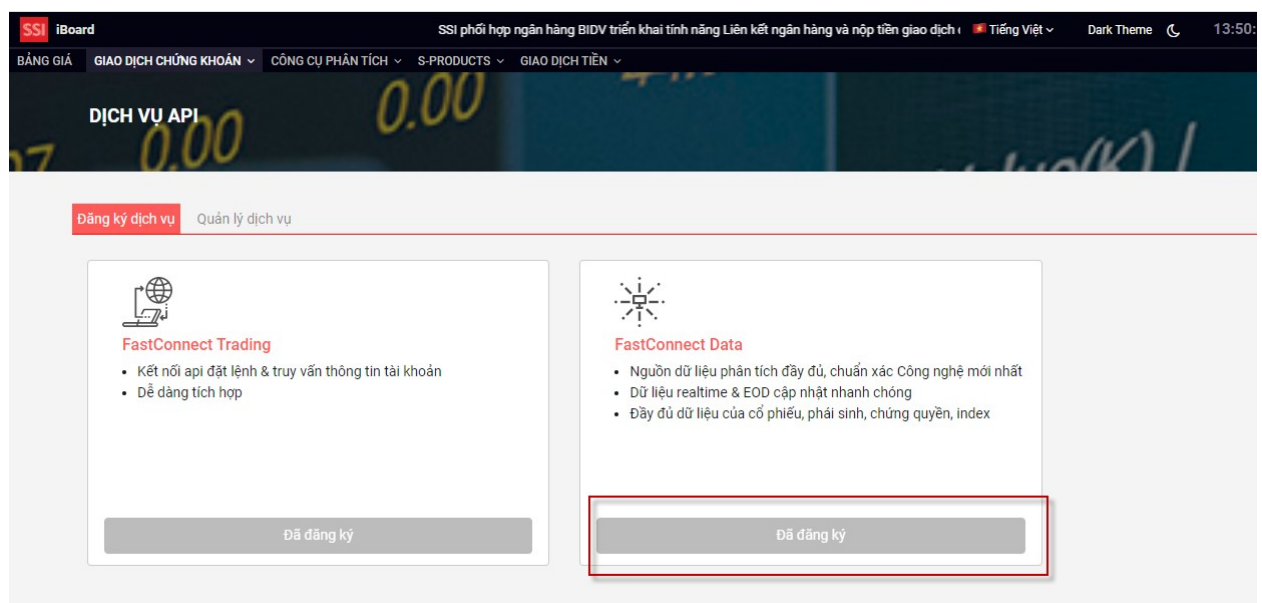
- Để tích hợp API FastConnect Trading cần:

+ Iboard:

Bước 1: Khách hàng đến các chi nhánh / PGD của SSI để đăng ký sử dụng dịch vụ Fast Connect API

Bước 2: Sau khi được thông báo đăng ký thành công, Khách hàng đăng nhập vào Iboard và vào tab DỊCH VỤ API.

Màn hình hiển thị đã đăng ký dịch vụ thành công:



Màn hình thông tin dịch vụ đã đăng ký:

SSI iBoard Đăng ký trước với lãi suất ưu đãi 0%/năm, dự nợ tối đa 2 tỷ đồng/TK. LH 1900545471 hoặc truy cập www.ssi.com.vn để biết thêm chi tiết. Tiếng Việt Dark Theme 13:54:51 Thứ Năm, 17/02/2022 50593@EARTH

DỊCH VỤ API

Đăng ký dịch vụ Quản lý dịch vụ

Tên dịch vụ	Kỳ hạn	Bắt đầu	Kết thúc	IP đăng ký	Trạng thái kết nối
✓ FCDData	3 tháng	30/12/2020	17/05/2022		Đang hoạt động
✓ FCTrading	3 tháng	30/12/2020	16/09/2022		Đang hoạt động
✓ QTrader					

Bước 3: Tạo Key kết nối để sử dụng dịch vụ:

SSI iBoard SSI phối hợp ngân hàng BIDV triển khai tính năng Liên kết Tiếng Việt Dark Theme 13:56:27 Thứ Năm, 17/02/2022 50593@EARTH

DỊCH VỤ API

Đăng ký dịch vụ Quản lý dịch vụ

Tên dịch vụ	Kỳ hạn	Bắt đầu	Kết thúc	IP đăng ký	Trạng thái kết nối
✓ FCDData	3 tháng	30/12/2020	17/05/2022	Bạn muốn tạo key kết nối cho	Đang hoạt động
✓ FCTrading	3 tháng	30/12/2020	16/09/2022		Đang hoạt động
✓ QTrader					

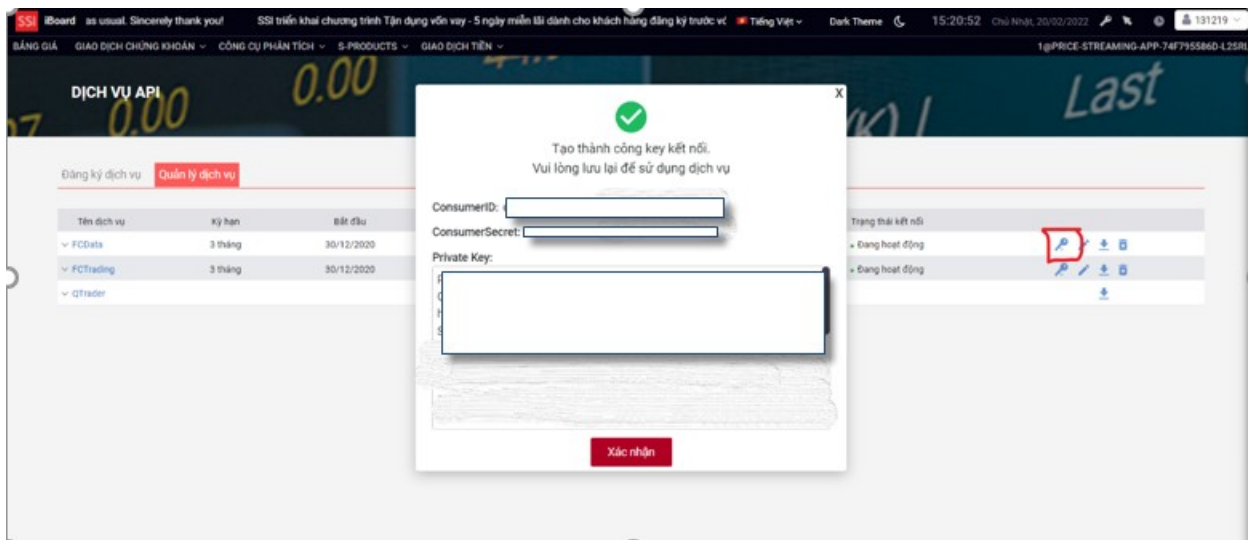
Dịch vụ FastConnect Data

PIN

Tạo key **Hủy**

(Key kết nối được tạo tối đa 3 lần/ngày)

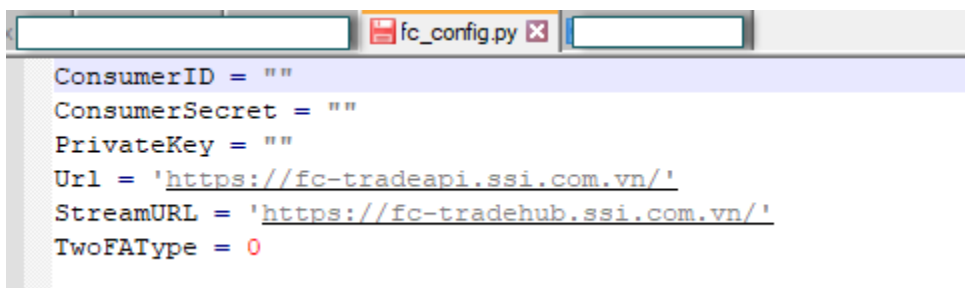
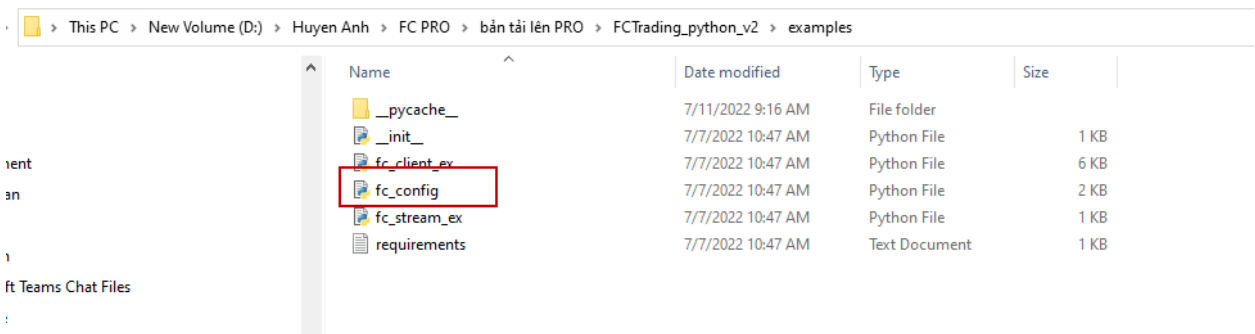
Key kết nối được tạo thành công:



Bước 4: Lấy ConsumerID và ConsumerSecret thay vào file config: fc_config (file trong thư mục giải nén khi tải client Python hoặc NodeJS Fastconnect Trading) và điền đầy đủ các thông tin:

+ TwoFAType: 0: PIN và 1: OTP

VD đối với client Python:



Bước 5: Lưu thông tin

1.2. Key Credential

Thông tin cấu hình để tích hợp với FCTradingAPI

- ConsumerID: Định danh tài khoản
- ConsumerSecret: Key truy cập server
- PrivateKey: Được sử dụng để tạo chữ ký số bằng thuật toán RS256

2. Thông tin tích hợp

2.1. Chữ ký điện tử

Signature là một chuỗi ký tự được tạo ra từ một thuật toán cho trước, sử dụng để kiểm tra tính đúng đắn của dữ liệu trên đường truyền giữa 2 hệ thống. Một số thuật toán đang sử dụng là MD5, SHA256, RSA.

Tham khảo thêm: [Wikipedia](#)

Trong tài liệu này sử dụng thuật toán RSA + SHA256 để tạo Signature. Dữ liệu đầu vào bao gồm Secret Key và data, data là một json body chứa thông tin đặt/sửa/xóa

2.2. Python Client

2.2.1. Cài đặt

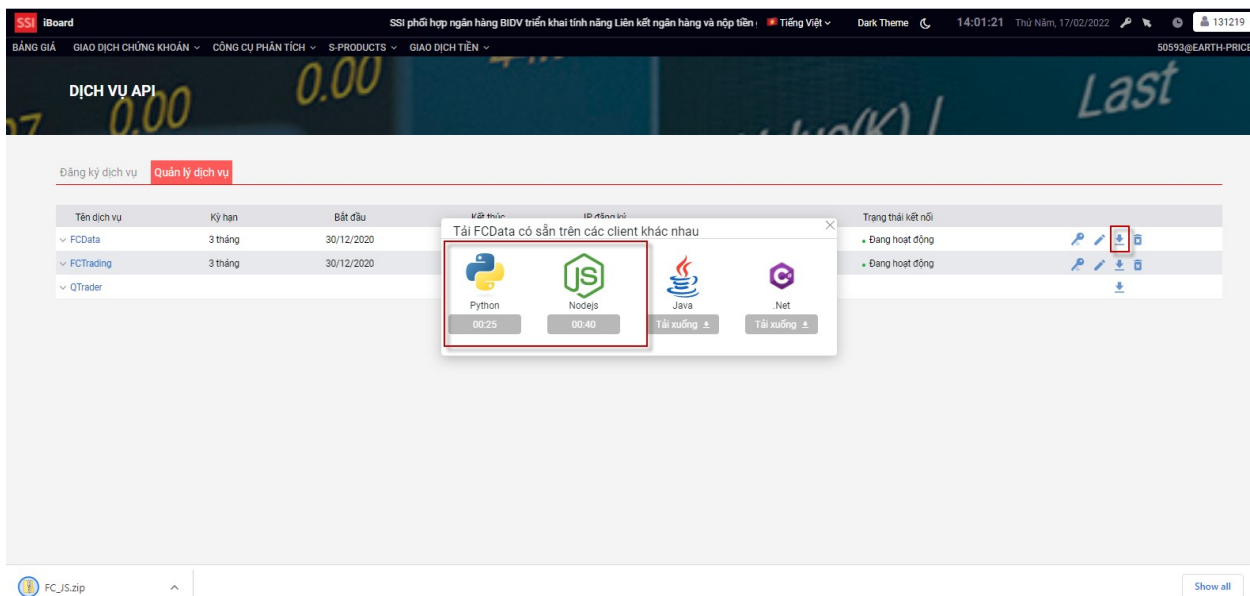
- Yêu cầu cài đặt Python 3.x trở lên (có thể tải bản python.org/downloads/release/python-370/)

Lưu ý: Nhớ tick Add system path khi setup Python

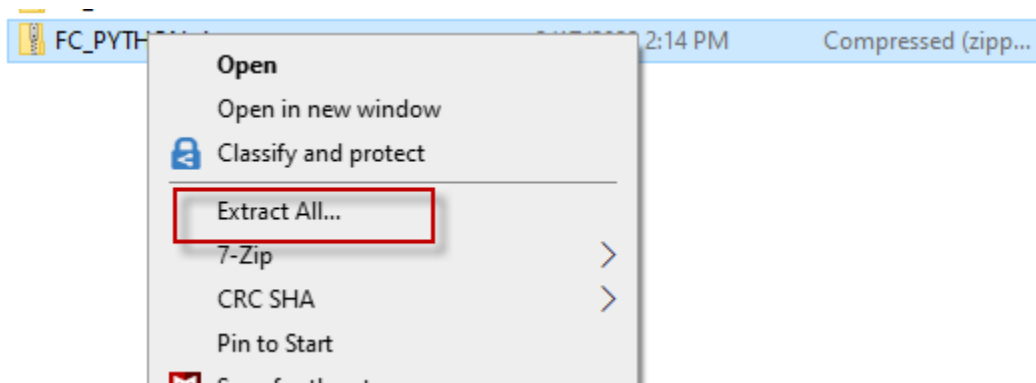
- Khi cài đặt không sử dụng phiên bản Python mới nhất

VD: Nếu phiên bản 3.9 là bản mới nhất thì tải bản 3.8 về về bản 3.0

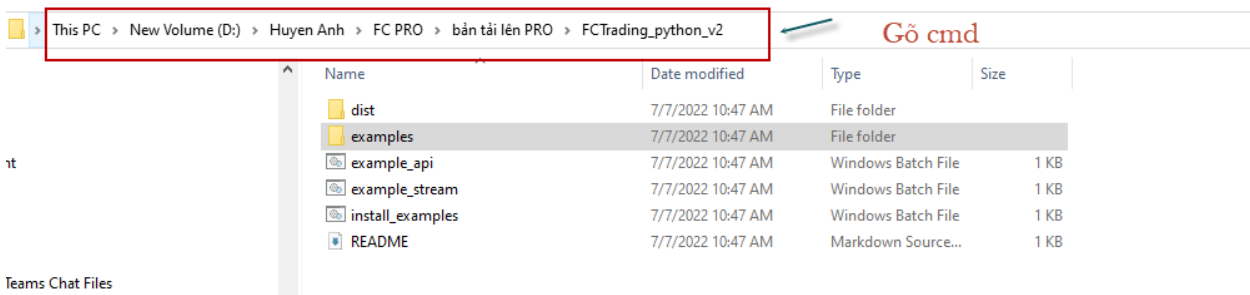
- Tải client và giải nén:



Bước 1: Giải nén



Bước 2: Gõ cmd vào ô địa chỉ thư mục để khởi tạo command line



Màn hình hiển thị cửa sổ như sau:

```
C:\Windows\System32\cmd.exe
Microsoft Windows [Version 10.0.19044.1706]
(c) Microsoft Corporation. All rights reserved.

D:\Huyen Anh\FC PRO\bản tải lên PRO\FCTrading_python_v2>_
```

Bước 3: Điền thông tin trong file `fc_config.py` tại thư mục `examples`:

```
ConsumerID = ""
ConsumerSecret = ""
PrivateKey = ""
Url = 'https://fc-tradeapi.ssi.com.vn/'
StreamURL = 'https://fc-tradehub.ssi.com.vn/'
TwoFAType = 0
```

Lưu ý: TwoFAType = 0: PIN và 1: OTP (SMS/ Email/ SmartOTP)

Mã OTP có thể chọn hiệu lực 8 tiếng, Khách hàng chỉ cần xác thực 1 lần đầu ngày là có thể sử dụng trong suốt phiên giao dịch.

Bước 4: Cài đặt Client: Dựa vào file README khi giải nén file:

```
# Run Example

- Install ssi-fctrading client: `pip install dist/ssi_fctrading-2.0.0.tar.gz`
- Install requirement: `pip install -r examples/requirements.txt`
- Config your key and PIN in file `examples/fc_config.py`
- Run `example_api.bat` to start example and open browser at [Swagger] (http://127.0.0.1:8000/docs) to test
- Run `example_stream.bat` to start streaming
```

Chạy câu lệnh sau: `pip install --trusted-host pypi.org --trusted-host files.pythonhosted.org --proxy=http://<username>:<password>@proxyhn.ssi.com.vn:9090 dist/<tên file trong thư mục dist>`

Lưu ý:

+ username và password (không có ký tự) đăng nhập vào lboard.

+ Nếu password có @ -> Chuyển thành %40

Ví dụ câu lệnh và kết quả trả như ảnh:

```
D:\Huyen Anh\FC PRO\ban tải lên PRO\FCTrading_python_v2>pip install --trusted-host pypi.org --trusted-host files.pythonh
osted.org --proxy=http://[redacted]@proxyhn.ssi.com.vn:9090 dist/ssi_fctrading-2.0.0.tar.gz
Processing d:\huyen anh\fc pro\ban tải lên pro\fctrading_python_v2\dist\ssi_fctrading-2.0.0.tar.gz
Requirement already satisfied: pycryptodome in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packages
(from ssi-fctrading==2.0.0) (3.15.0)
Collecting xmljson
  Downloading xmljson-0.2.1-py2.py3-none-any.whl (10 kB)
Requirement already satisfied: requests in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packages (fr
om ssi-fctrading==2.0.0) (2.28.1)
Requirement already satisfied: signalr-client in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packag
e
s (from ssi-fctrading==2.0.0) (0.0.7)
Collecting pyjwt
  Downloading PyJWT-2.4.0-py3-none-any.whl (18 kB)
```

```
C:\Windows\System32\cmd.exe
te-packages (from requests->ssi-fctrading==2.0.0) (2.1.0)
Requirement already satisfied: sseclient in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packages (fr
om signalr-client->ssi-fctrading==2.0.0) (0.0.27)
Requirement already satisfied: gevent in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packages (from
signalr-client->ssi-fctrading==2.0.0) (21.12.0)
Requirement already satisfied: websocket-client in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packa
ges (from signalr-client->ssi-fctrading==2.0.0) (1.3.3)
Requirement already satisfied: six in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packages (from sse
client->signalr-client->ssi-fctrading==2.0.0) (1.16.0)
Requirement already satisfied: cffi>=1.12.2; platform_python_implementation == "CPython" and sys_platform == "win32" in
c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packages (from gevent->signalr-client->ssi-fctrading==2.
0.0) (1.15.1)
Requirement already satisfied: greenlet<2.0,>=1.1.0; platform_python_implementation == "CPython" in c:\users\anhnh2\appd
ata\local\programs\python\python39\lib\site-packages (from gevent->signalr-client->ssi-fctrading==2.0.0) (1.1.2)
Requirement already satisfied: zope.event in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packages (f
rom gevent->signalr-client->ssi-fctrading==2.0.0) (4.5.0)
Requirement already satisfied: zope.interface in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packag
es (from gevent->signalr-client->ssi-fctrading==2.0.0) (5.4.0)
Requirement already satisfied: setuptools in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packages (f
rom gevent->signalr-client->ssi-fctrading==2.0.0) (49.2.1)
Requirement already satisfied: pycparser in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packages (fr
om cffi>=1.12.2; platform_python_implementation == "CPython" and sys_platform == "win32"->gevent->signalr-client->ssi-fc
trading==2.0.0) (2.21)
Using legacy 'setup.py install' for ssi-fctrading, since package 'wheel' is not installed.
Installing collected packages: xmljson, pyjwt, ssi-fctrading
  Running setup.py install for ssi-fctrading ... done
Successfully installed pyjwt-2.4.0 ssi-fctrading-2.0.0 xmljson-0.2.1
WARNING: You are using pip version 20.2.3; however, version 22.1.2 is available.
You should consider upgrading via the 'c:\users\anhnh2\appdata\local\programs\python\python39\python.exe -m pip install
--upgrade pip' command.
```

Bước 5: Chạy tiếp câu lệnh pip install --trusted-host pypi.org --trusted-host

files.pythonhosted.org --proxy http://<username>:<password>@proxyhn.ssi.com.vn:9090 -r
examples/requirements.txt

Lưu ý: Nếu password có @ -> Chuyển thành %40

Ví dụ câu lệnh và kết quả như ảnh:

```
C:\Windows\System32\cmd.exe

D:\Huyen Anh\FC PRO\bản tải lên PRO\FCTrading_python_v2>pip install --trusted-host pypi.org --trusted-host files.pythonh
osted.org --proxy=http://[REDACTED]:[REDACTED]@proxyhn.ssi.com.vn:9090 -r examples/requirements.txt
Collecting fastapi==0.68.0
  Downloading fastapi-0.68.0-py3-none-any.whl (52 kB)
    | 52 kB 199 kB/s
Collecting uvicorn[standard]==0.14.0
  Downloading uvicorn-0.14.0-py3-none-any.whl (50 kB)
    | 50 kB 1.6 MB/s
Collecting starlette==0.14.2
  Downloading starlette-0.14.2-py3-none-any.whl (60 kB)
    | 60 kB 1.9 MB/s
Requirement already satisfied: pydantic!=1.7,!=1.7.1,!=1.7.2,!=1.7.3,!=1.8,!=1.8.1,<2.0.0,>=1.6.2 in c:\users\anhnh2\app
data\local\programs\python\python39\lib\site-packages (from fastapi==0.68.0->-r examples/requirements.txt (line 1)) (1.9
.1)
Collecting h11>=0.8
  Downloading h11-0.13.0-py3-none-any.whl (58 kB)
    | 58 kB 2.6 MB/s
Collecting click>=7.*
  Downloading click-8.1.3-py3-none-any.whl (96 kB)
    | 96 kB 1.3 MB/s
Collecting asgiref>=3.3.4
  Downloading asgiref-3.5.2-py3-none-any.whl (22 kB)
Collecting python-dotenv>=0.13; extra == "standard"
  Downloading python_dotenv-0.20.0-py3-none-any.whl (17 kB)
Collecting watchdog>=0.6; extra == "standard"
  Downloading watchdog-0.8.2-py3-none-any.whl (12 kB)
Collecting PyYAML>=5.1; extra == "standard"
  Downloading PyYAML-6.0-cp39-cp39-win_amd64.whl (151 kB)
    | 151 kB 3.2 MB/s
```

```
C:\Windows\System32\cmd.exe

Collecting websockets>=0.1; extra == "standard"
  Downloading websockets-10.3-cp39-cp39-win_amd64.whl (98 kB)
    | 98 kB 6.4 MB/s
Collecting colorama>=0.4; sys_platform == "win32" and extra == "standard"
  Downloading colorama-0.4.5-py2.py3-none-any.whl (16 kB)
Requirement already satisfied: typing-extensions>=3.7.4.3 in c:\users\anhnh2\appdata\local\programs\python\python39\lib\
site-packages (from pydantic!=1.7,!=1.7.1,!=1.7.2,!=1.7.3,!=1.8,!=1.8.1,<2.0.0,>=1.6.2->fastapi==0.68.0->-r examples/req
uirements.txt (line 1)) (4.3.0)
Collecting anyio<4,>=3.0.0
  Downloading anyio-3.6.1-py3-none-any.whl (80 kB)
    | 80 kB 5.1 MB/s
Requirement already satisfied: idna>=2.8 in c:\users\anhnh2\appdata\local\programs\python\python39\lib\site-packages (fr
om anyio<4,>=3.0.0->watchdog>=0.6; extra == "standard"->uvicorn[standard]==0.14.0->-r examples/requirements.txt (line 2)
) (3.3)
Collecting sniffio>=1.1
  Downloading sniffio-1.2.0-py3-none-any.whl (10 kB)
Installing collected packages: starlette, fastapi, h11, colorama, click, asgiref, python-dotenv, sniffio, anyio, watchgo
d, PyYAML, httptools, websockets, uvicorn
Successfully installed PyYAML-6.0 anyio-3.6.1 asgiref-3.5.2 click-8.1.3 colorama-0.4.5 fastapi-0.68.0 h11-0.13.0 httptoo
ls-0.2.0 python-dotenv-0.20.0 sniffio-1.2.0 starlette-0.14.2 uvicorn-0.14.0 watchdog-0.8.2 websockets-10.3
WARNING: You are using pip version 20.2.3; however, version 22.1.2 is available
You should consider upgrading via the 'c:\users\anhnh2\appdata\local\programs\python\python39\python.exe -m pip install
--upgrade pip' command.

D:\Huyen Anh\FC PRO\bản tải lên PRO\FCTrading_python_v2>
```


Bước 6: Click double vào file example_api -> Cài đặt thành công như ảnh:

[illegible]

Note:

- Chỉ cài đặt 1 lần duy nhất. Lần sau chạy chỉ cần click double vào file example_api và Đặt/sửa/hủy lệnh và truy vấn thông tin tại link Swagger:

<http://127.0.0.1:8000/docs>

2.2.2. Chạy Client

Mở link Swagger đính kèm trong file README và thực hiện truy vấn/ đặt lệnh... trực tiếp trên Swagger: <http://127.0.0.1:8000/docs>

FastAPI

0.1.0

OAS3

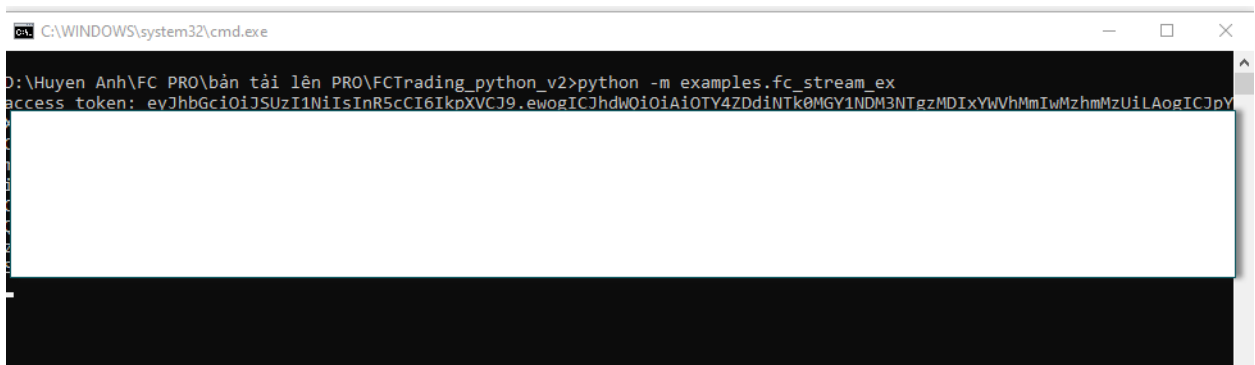
/openapi.json

default

GET	/getOtp	Fc Get Otp	
GET	/verifyCode	Fc Get Otp	
GET	/newOrder	Fc New Order	
GET	/modifyOrder	Fc Modify Order	
GET	/cancelOrder	Fc Cancel Order	
GET	/stockAccountBalance	Fc Stock Account Balance	
GET	/derivativeAccountBalance	Fc Derivative Account Balance	
GET	/ppmmrAccount	Fc Pp Mmr Account	
GET	/stockPosition	Fc Stock Position	
GET	/derivativePosition	Fc Derivative Position	

2.2.3. Streaming

- Click double vào file example_stream để nghe Streaming lệnh. Thông tin như ảnh:

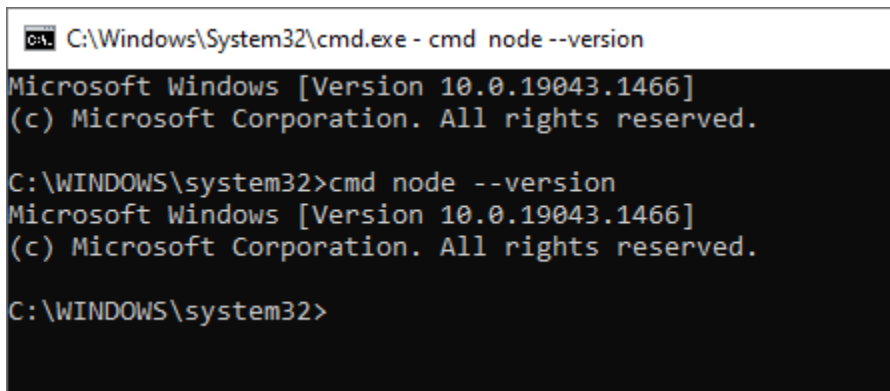


```
C:\WINDOWS\system32\cmd.exe
D:\Huyen Anh\FC PRO\bản tải lên PRO\FCTrading_python_v2>python -m examples.fc_stream_ex
access token: eyJhbGciOiJSUzI1NiIsInR5cCI6IkpXVCJ9.ewogICJhdWoiOiAiOTY4ZDdiNTk0MGY1NDM3NTgzMDIxYVhMmIwMzhhMzUiLAogICJpY
```

2.3. NodeJS Client

2.3.1. Cài đặt

- Máy tính có cài đặt NodeJS (cài đặt từ bản v.10.15.3 trở lên)
- Kiểm tra máy tính đã cài đặt NodeJS và version: cmd node --version

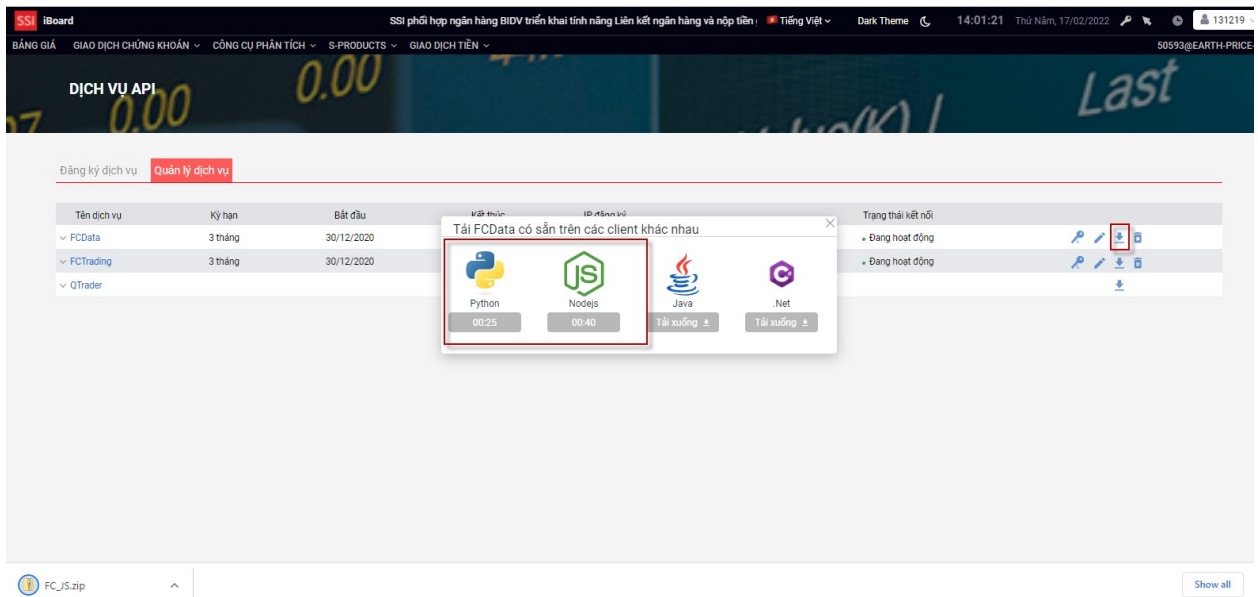


```
C:\Windows\System32\cmd.exe - cmd node --version
Microsoft Windows [Version 10.0.19043.1466]
(c) Microsoft Corporation. All rights reserved.

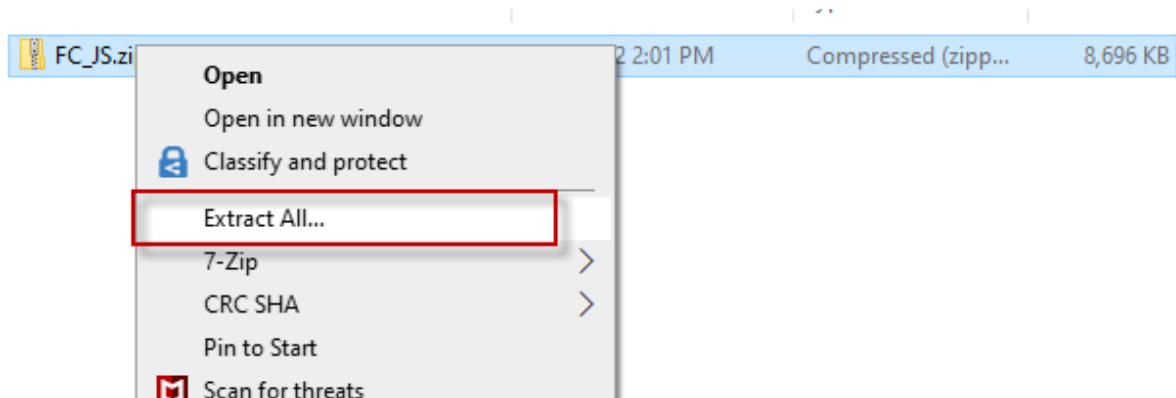
C:\WINDOWS\system32>cmd node --version
Microsoft Windows [Version 10.0.19043.1466]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>
```

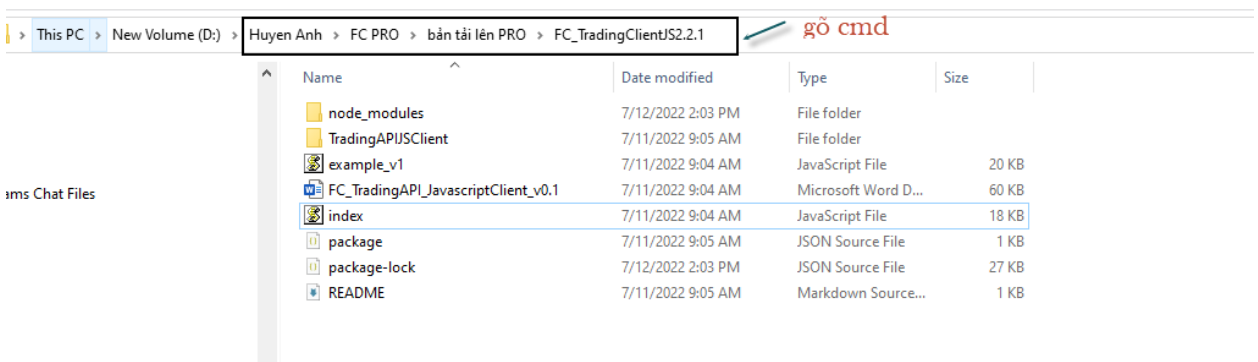
- Tải client và giải nén:



Bước 1: Giải nén



Bước 2: Gõ cmd vào ô địa chỉ thư mục để khởi tạo command line



Màn hình hiển thị cửa sổ như sau:

```
C:\Windows\System32\cmd.exe

Microsoft Windows [Version 10.0.19043.1466]
(c) Microsoft Corporation. All rights reserved.

F:\tài liệu\PRO\FCtrading\FC_JS\FCTrading_JS_v2.0.1>
```

Bước 3: Khi chạy xong tiếp tục gõ câu lệnh: node index.js -> Enter

```
C:\Windows\System32\cmd.exe - node index.js

F:\tài liệu\PRO\FCtrading\FC_JS\FCTrading_JS_v2.0.1>node index.js
Example app listening on port 3012!
Please specify a ConsumerID; Please specify a ConsumerSecret
_
```

Bước 4: Giải nén client thành công:

```
C:\Windows\System32\cmd.exe - node index.js

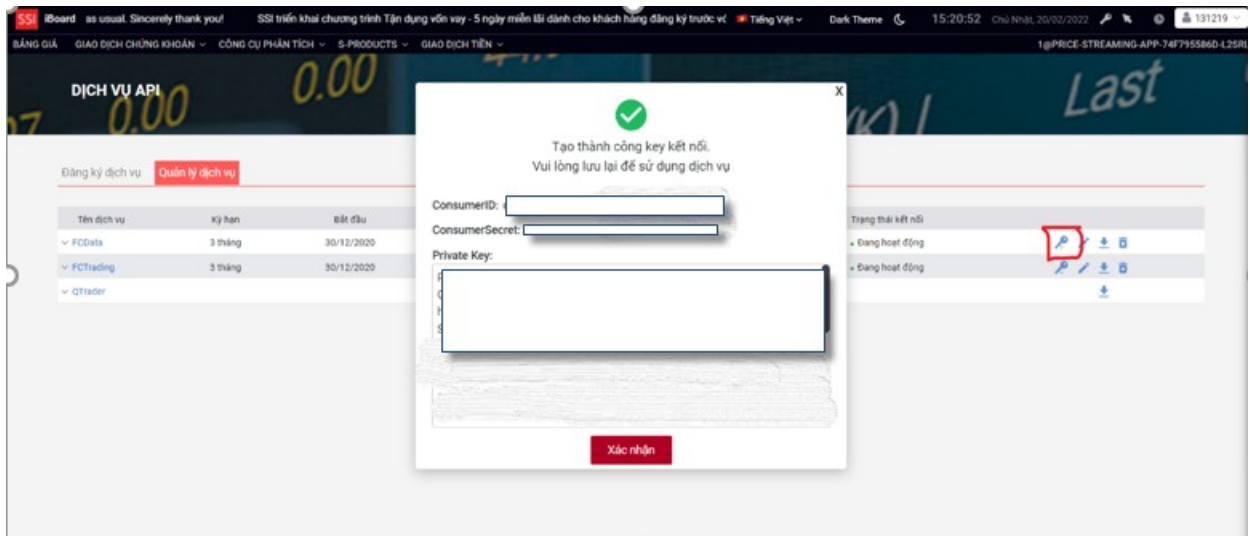
F:\tài liệu\PRO\FCtrading\FC_JS\FCTrading_JS_v2.0.1>node index.js
Example app listening on port 3012!
Please specify a ConsumerID; Please specify a ConsumerSecret
_
```

2.3.2. Chạy Client

- Để thuận tiện cho việc test dữ liệu từ Client, người dùng có thể sử dụng thông tin được điền sẵn trong file example_v1.js.

- Lấy Key được tạo ở Iboard thay vào file config.js và chạy lại cmd như sau:

Bước 1: Tạo Key ở IBoard:



Bước 2:

- Thay ConsumerID và ConsumerSecret vào file index.js:



Bước 3: Phương thức giao dịch của khách hàng

Bước 3.1: Khách hàng sử dụng PIN để giao dịch:

- Điền PIN khách hàng giao dịch vào trường Code ở 2 hàm "mockStockData" và "mockDeterativeData"

```
ex.js x index.js x
var date = new Date();

//This is config for consumer have permission on all customer
var config = {
  ConsumerID:
  ConsumerSec:
  PrivateKey:

  URL: "https://fc-tradeapi.ssi.com.vn",
  stream_url: "wss://fc-tradehub.ssi.com.vn/",
};
var configServer = config;
const rq = axios.create({
  baseURL: config.URL,
  timeout: 5000
})
/** @END CONFIG */
var mockStockData = {
  account: "0303961",
  buySell: "B",
  market: "VN", // Only support "VN" and "VNFE"
  ordertype: "LO",
  price: 25000,
  quantity: 300,
  instrumentid: "SSI",
  validitydate: date.yyyymmdd(),
  channel: "IW",
  extOrderID: "", // this property is unique in day.
  session: ""
  code:
  twofaType: 0,
  startDate: "24/05/2019",
  endDate: "30/05/2019"
}
```

➔ Lưu file index.js

Bước 4: Mở cmd và chạy lại câu lệnh: ctrl C -> node index.js

```
C:\Windows\System32\cmd.exe - node index.js
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.

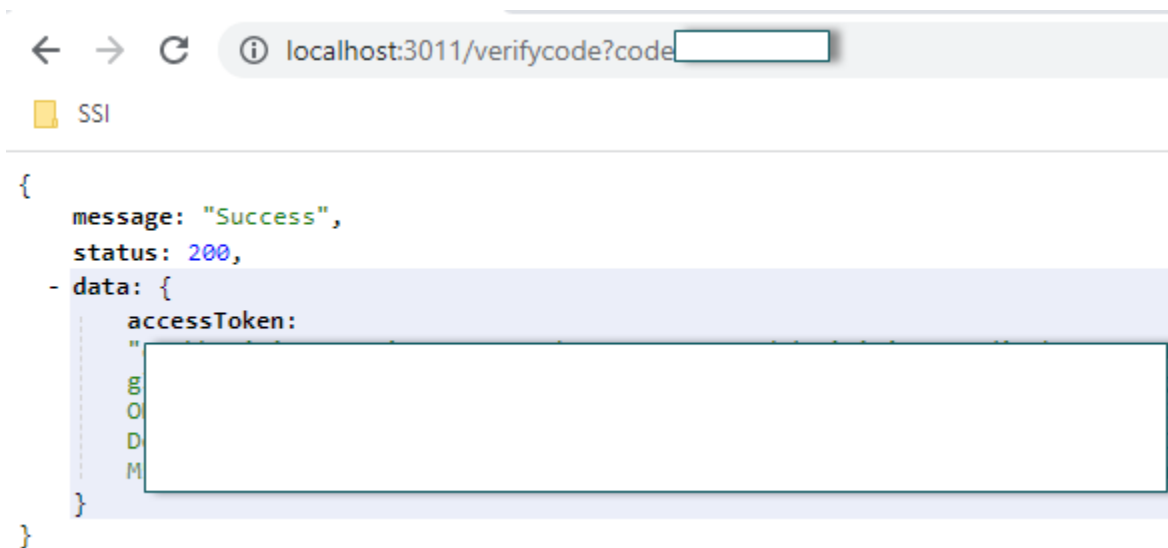
D:\Huyen Anh\FC PRO\FC 2FA\FCTradingClient_v2.3>node index.js
Example app listening on port 3011!
{
```

Bước 5: Giải nén và Cài đặt client thành công

```
C:\Windows\System32\cmd.exe - node index.js
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.

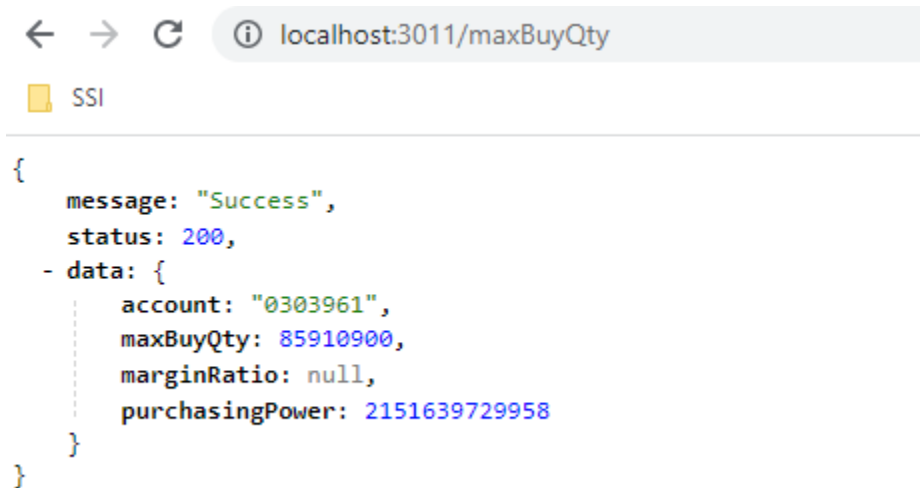
D:\Huyen Anh\FC PRO\FC 2FA\FCTradingClient_v2.3>node index.js
Example app listening on port 3011!
{
  message: 'Success',
  status: 200,
  data: {
    accessToken: 'eyJhbGciOiJIUzUuIiwiaWQiOiJlbnR5cCI6IkpXVCJ9.ewogICJhdwQiOiAiOTY4ZDd1NTk0MGY1NDM3NTgzMDIxYWVhMmIwMzhmMzU1LAogI
  }
}
Connected!
orderError:
{"type":"orderError","data":{"message":"Order price lower limit can not exceed spread limit","notifyID":95,"errorCode":"ORD002","uniqueID":"72711796","ipAddress":"10.48.41.15","prefix":"89p"}}}
```

Bước 6: Chạy trình duyệt hàm “/verifyCode” để verify mã PIN:



Bước 7: Đặt/ Hủy/ Sửa lệnh và Query thông tin lệnh như trước đây:

Ví dụ như ảnh:

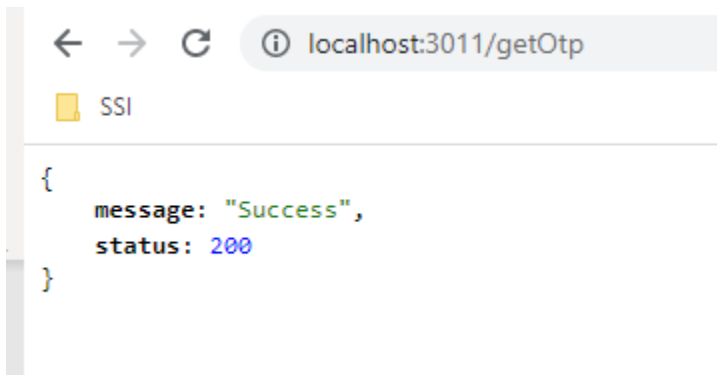


```
{
  message: "Success",
  status: 200,
  data: {
    account: "0303961",
    maxBuyQty: 85910900,
    marginRatio: null,
    purchasingPower: 2151639729958
  }
}
```

Bước 3.2. Khách hàng sử dụng 2FA để giao dịch (SMS/ Email/ SmartOTP)

- Chạy trình duyệt hàm “/getOtp” để lấy mã OTP ở điện thoại nếu là SMS/ SmartOTP hoặc OTP ở email nếu là Email

Note: Mã OTP có thể chọn hiệu lực 8 tiếng, Khách hàng chỉ cần xác thực 1 lần đầu ngày là có thể sử dụng trong suốt phiên giao dịch.



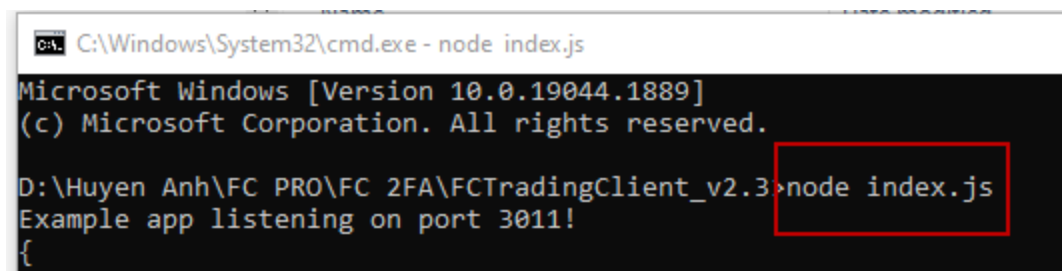
```
{
  message: "Success",
  status: 200
}
```

- Điền mã OTP nhận được vào trường code của 2 hàm “mockStockData” và “mockDeterativeData”


```
/** @END_CONFIG */  
var mockStockData = {  
  account: "0303961",  
  buySell: "B",  
  market: "VN", // Only support "VN" and "VNFE"  
  ordertype: "LO",  
  price: 25000,  
  quantity: 300,  
  instrumentid: "SSI",  
  validitydate: date.yyyymmdd(),  
  channel: "IW",  
  extOrderID: "", // this property is unique in day.  
  session: "",  
  code: "",  
  twoFaType: "0",  
  startDate: "24/05/2019",  
  endDate: "30/05/2019"  
};
```

→ Lưu file index.js

Bước 4: Mở cmd và chạy lại câu lệnh: ctrl C -> node index.js



C:\Windows\System32\cmd.exe - node index.js
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.
D:\Huyen Anh\FC PRO\FC 2FA\FCTradingClient_v2.3>node index.js
Example app listening on port 3011!
{

Bước 5: Giải nén và Cài đặt client thành công

```
C:\Windows\System32\cmd.exe - node index.js
Microsoft Windows [Version 10.0.19044.1889]
(c) Microsoft Corporation. All rights reserved.

D:\Huyen Anh\FC PRO\FC 2FA\FCTradingClient_v2.3>node index.js
Example app listening on port 3011!
{
  message: 'Success',
  status: 200,
  data: {
    accessToken: 'eyJhbGciOiJIUzUuIiwiaWQiOiJ0OTY4ZDd1NTk0MGY1NDM3NTgzMDIxYWVhMmIwMzhhMzU1LAogI
}
}
Connected!
orderError:
{"type":"orderError","data":{"message":"Order price lower limit can not exceed spread limit","notifyID":95,"errorCode":"ORD002","uniqueID":"72711796","ipAddress":"10.48.41.15","prefix":"89p"}}
}
```

Bước 6: Đặt/ Hủy/ Sửa lệnh và Query thông tin lệnh như trước đây:

Ví dụ như ảnh:

```
← → ↻ ⓘ localhost:3011/maxBuyQty
SSI
{
  message: "Success",
  status: 200,
  - data: {
    account: "0303961",
    maxBuyQty: 85910900,
    marginRatio: null,
    purchasingPower: 2151639729958
  }
}
```

2.3.3. Market data

a. Streaming

Streaming trading chỉ có khi đặt/ sửa/ hủy lệnh thành công

[illegible]

b. API

- Mở file main.js (bằng Notepad ++...) và nhập thông tin cần kiểm tra

Bước 1: Điền thông tin hàm muốn truy vấn ở file index.js

Bước 2: Lưu file và chạy lại cmd theo câu lệnh:

<http://localhost:<port>hienthitrongcmd/<tenhamtruyvan>>

Bước 3: Kiểm tra thông tin vừa truy vấn

Ví dụ: Lấy thông tin Get Max Buy Qty

- Điền thông tin muốn truy vấn ở hàm Maxbuyqty trong file index.js -> Save

```
index.js
494     }).catch(error => {
495       res.send(error);
496     })
497   });
498   app.get("/accountBalance", (req, res) => {
499     var ro = {};
500     Object.assign(ro, mockStockData);
501     Object.assign(ro, req.query);
502     var request = {
503       account: 1312191
504     }
505     rq({
506       url: client.api.GET_ACCOUNT_BALANCE,
507       method: 'get',
```

- Mở trình duyệt và chạy câu lệnh: `http://localhost:3012/accountBalance`

```
{
  message: "Success",
  status: 200,
  data: {
    account: "1312191",
    cashBal: 29573,
    cashOnHold: 0,
    secureAmount: 0,
    withdrawable: 29573,
    receivingCashT1: 0,
    receivingCashT2: 0,
    matchedBuyVolume: 0,
    matchedSellVolume: 0,
    debt: 0,
    unmatchedBuyVolume: 0,
    unmatchedSellVolume: 0,
    paidCashT1: 0,
    paidCashT2: 0,
    cia: 0,
    purchasingPower: 29573,
    totalAssets: 29573
  }
}
```