

# HƯỚNG DẪN SỬ DỤNG FASTCONNECT DATA

FastConnect Data (FCData) là bộ API cung cấp dữ liệu thị trường và thông tin cơ bản về các mã chứng khoán và các index. Khách hàng có thể lấy dữ liệu EOD hoặc realtime tùy nhu cầu sử dụng. SSI hỗ trợ môi trường Python, NodeJS

## 1. Quy trình tích hợp

Để tích hợp FC Data, User cần đăng ký sử dụng dịch vụ trên Iboard (<https://iboard.ssi.com.vn>) hoặc tại quầy giao dịch chứng khoán.

### 1.1. Đăng ký dịch vụ

- Để tích hợp API FastConnect Data cần:

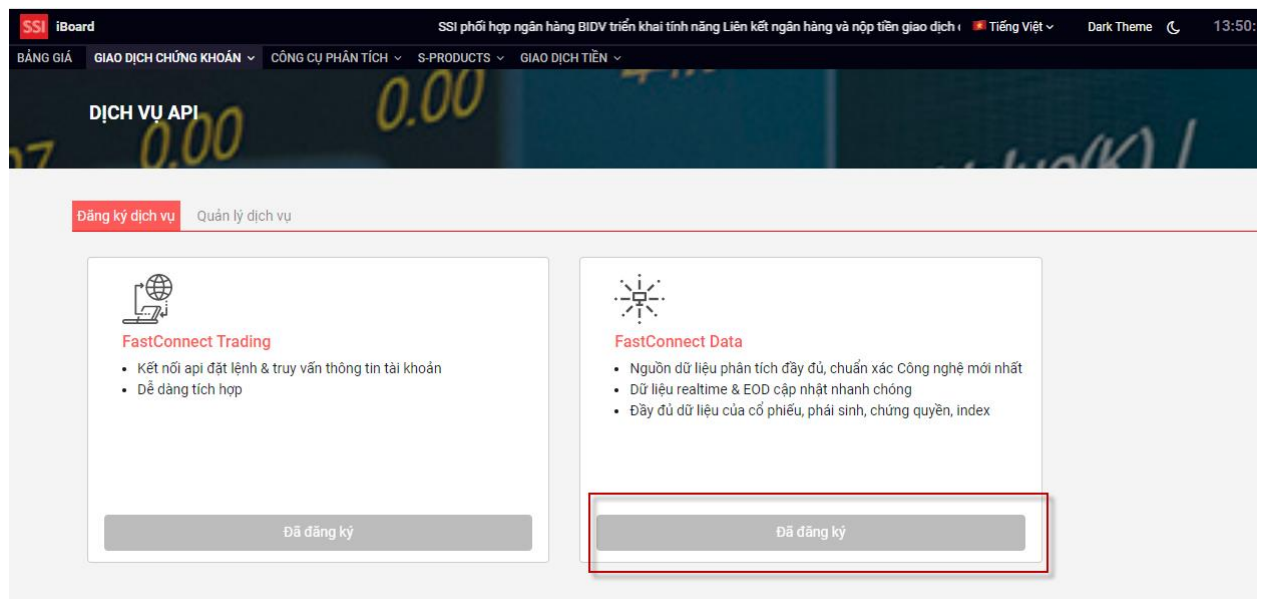
+ Iboard:

Bước 1: Người dùng đăng nhập [iboard.ssi.com.vn](https://iboard.ssi.com.vn) bằng tài khoản cá nhân. Chọn GIAO DỊCH CHỨNG KHOÁN -> DỊCH VỤ API.

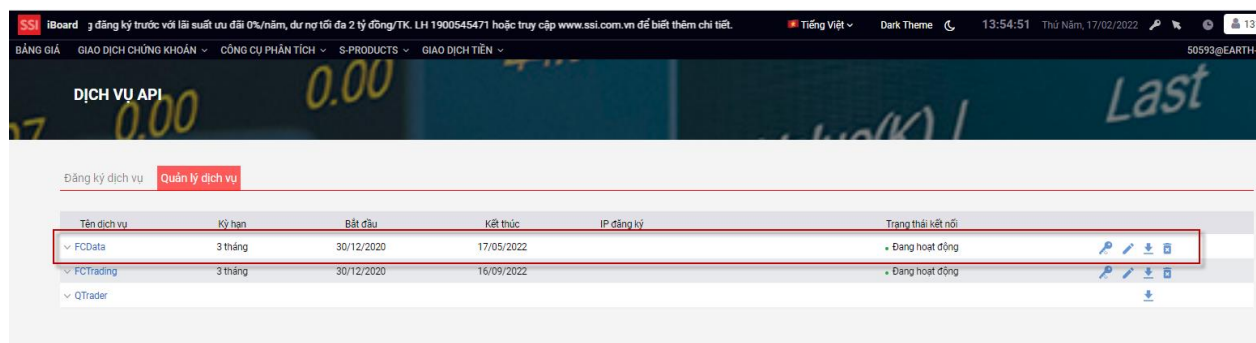
Bước 2: Đăng ký dịch vụ:

Bước 3: Sau khi đăng ký thành công, hệ thống sẽ gửi email đến người dùng và giao diện trên Iboard:

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

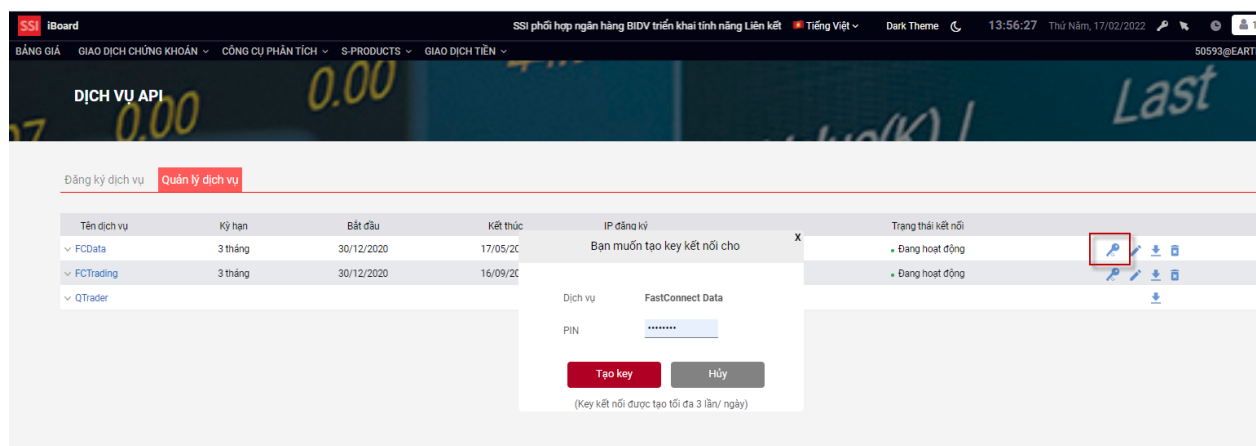


Màn hình thông tin dịch vụ đã đăng ký:



Tên dịch vụ	Kỳ hạn	Bắt đầu	Kết thúc	IP đăng ký	Trạng thái kết nối
FCData	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 4: Tạo Key kết nối để sử dụng dịch vụ:



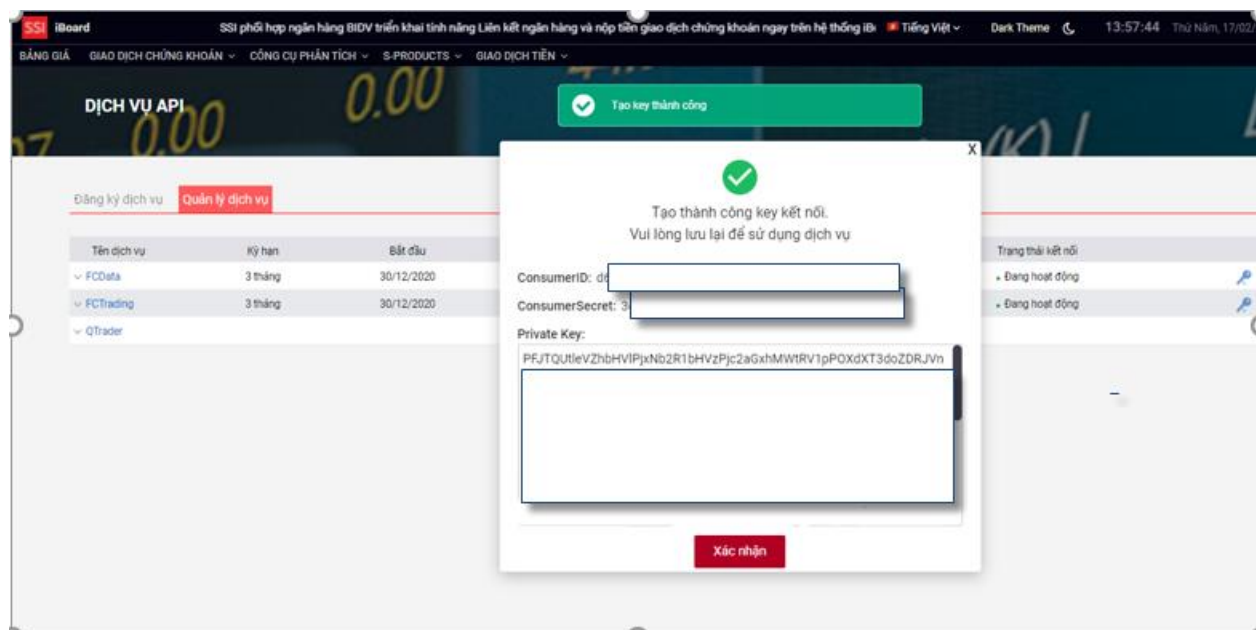
Tên dịch vụ: FCData, Kỳ hạn: 3 tháng, Bắt đầu: 30/12/2020, Kết thúc: 17/05/2022, IP đăng ký: Ban muốn tạo key kết nối cho, Trạng thái kết nối: Đang hoạt động

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:



Tạo key thành công

Tạo thành công key kết nối.  
Vui lòng lưu lại để sử dụng dịch vụ

ConsumerID: d4...  
ConsumerSecret: 3...  
Private Key: PFJtQUTleVZhbHVPjxNb2R1bHVzPjc2a5xhMWlRV1pPOXdXT3doZDRJVn

Xác nhận

Bước 5: Lấy ConsumerID và ConsumerSecret thay vào file config: config.py ( file trong thư mục giải nén khi tải client Python hoặc NodeJS Fastconnect Data)

VD đối với client Python:

his PC > New Volume (F:) > tài liệu > PRO > FCdata > FC\_PYTHON > FCData\_PY

Name	Date modified	Type	Size
__pycache__	2/17/2022 2:18 PM	File folder	
config.py	1/11/2022 2:41 PM	PY File	1 KB
ssi_fc_data-2.0.0.tar.gz	11/18/2020 11:53 AM	GZ File	9 KB
test_Req_Res.py	12/15/2020 5:32 PM	PY File	2 KB
test_Streaming.py	12/15/2020 5:32 PM	PY File	1 KB

```
1 auth_type = 'Bearer'
2
3
4 consumerID = ''
5 consumerSecret = ''
6
7 access_jwt = ''
8
9
10 url = 'https://fc-data.ssi.com.vn/'
11 stream_url = 'https://fc-data.ssi.com.vn/'
```

Bước 6: Lấy access\_jwt khi chạy cmd. Đọc hướng dẫn ở mục 2. Thông tin tích hợp  
-> 2.1.1. Cài đặt

Bước 7: Lưu thông tin

## 2. Thông tin tích hợp

### 2.1. Python Client

#### 2.1.1. Cài đặt

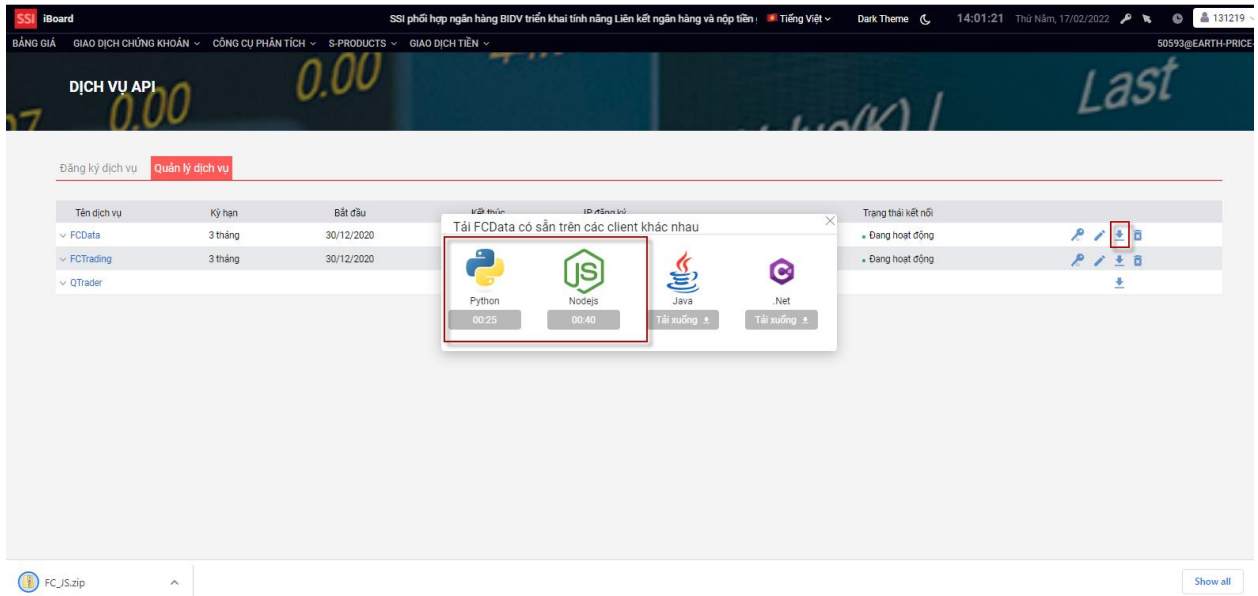
- Yêu cầu cài đặt Python 3.x trở lên ( có thể tải bản [python.org/downloads/release/python-370/](https://python.org/downloads/release/python-370/))

Lưu ý: Nhớ tick Add system path khi setup Python

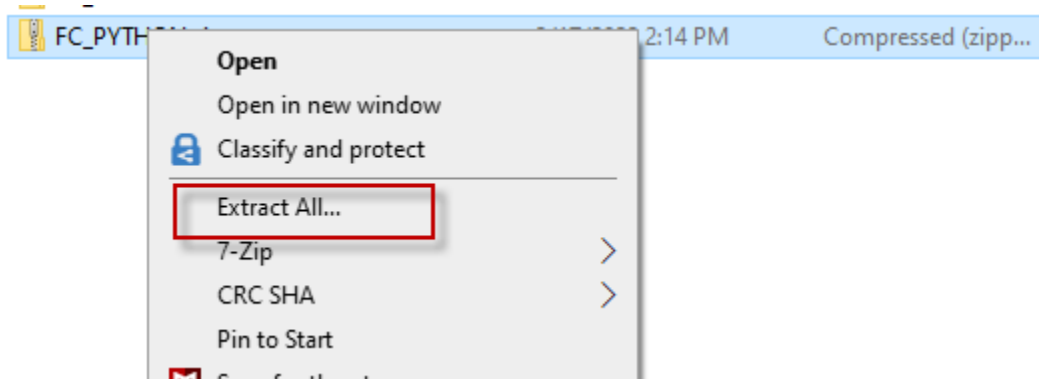
- Khi cài đặt không sử dụng phiên bản Python mới nhất

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

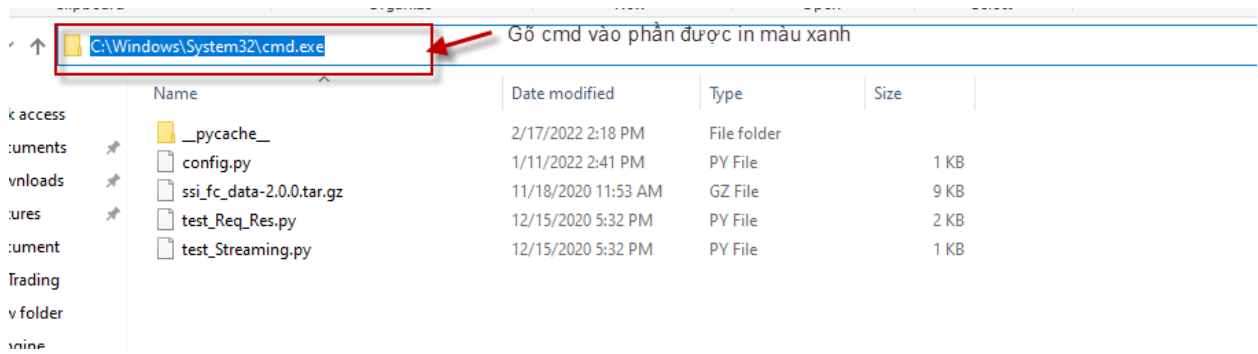
- Tải client và giải nén:



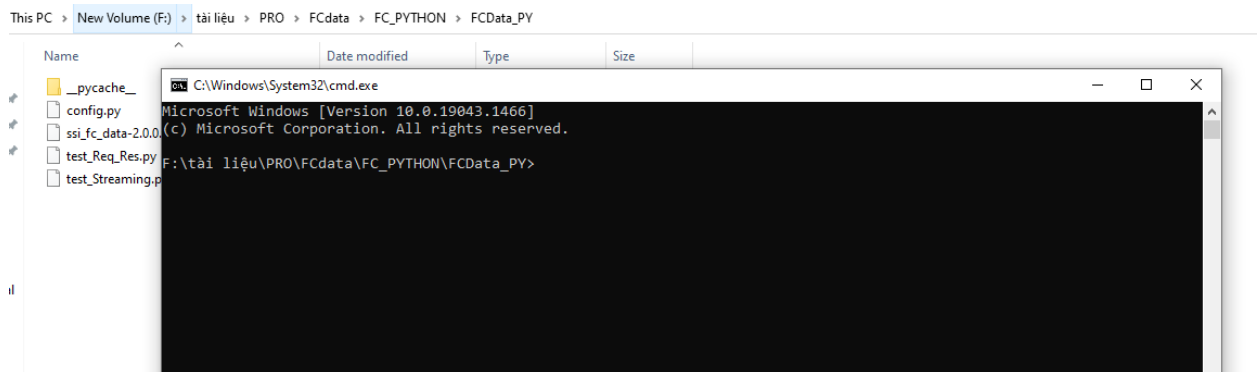
Bước 1: Giải nén



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



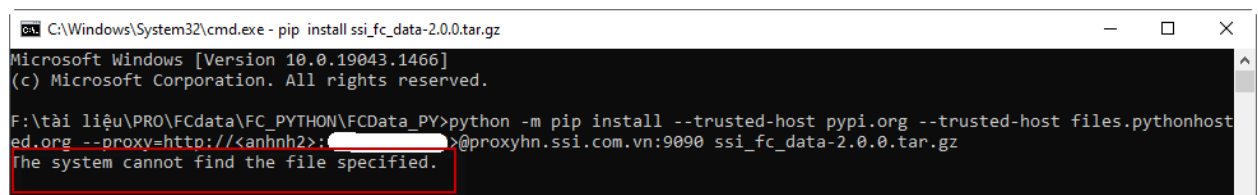
Màn hình hiển thị cửa sổ như sau:



Bước 3: Cài đặt Client theo câu lệnh sau: `python -m pip install --trusted-host pypi.org --trusted-host files.pythonhosted.org --proxy=http://<username>:<password>@proxyhn.ssi.com.vn:9090 <đường dẫn đến file>`

Lưu ý: username và password (không có ký tự) đăng nhập vào Iboard.

Ví dụ như ảnh sau:



Bước 4: Nếu trả kết quả khoảng như ảnh -> Máy không có proxy nên chỉ cần chạy câu lệnh: `pip install <đường dẫn đến file>`

Màn hình trả kết quả như ảnh:

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

F:\tài liệu\PRO\FCdata\FC_PYTHON\FCData_PY>pip install ssi_fc_data-2.0.0.tar.gz
Processing f:\tài liệu\pro\fcdata\fc_python\fcdata_py\ssi_fc_data-2.0.0.tar.gz
Requirement already satisfied: requests in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from ssi-fc-data==2.0.0) (2.25.0)
Requirement already satisfied: signalr-client in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from ssi-fc-data==2.0.0) (0.0.7)
Requirement already satisfied: chardet<4,>=3.0.2 in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from requests->ssi-fc-data==2.0.0) (3.0.4)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from requests->ssi-fc-data==2.0.0) (2020.11.8)
Requirement already satisfied: idna<3,>=2.5 in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from requests->ssi-fc-data==2.0.0) (2.10)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from requests->ssi-fc-data==2.0.0) (1.26.2)
Requirement already satisfied: websocket-client in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from signalr-client->ssi-fc-data==2.0.0) (0.57.0)
Requirement already satisfied: gevent in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from signalr-client->ssi-fc-data==2.0.0) (20.9.0)
Requirement already satisfied: sseclient in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from signalr-client->ssi-fc-data==2.0.0) (0.0.27)
Requirement already satisfied: greenlet>=0.4.17 in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from gevent->signalr-client->ssi-fc-data==2.0.0) (0.4.17)
Requirement already satisfied: cffi>=1.12.2 in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from gevent->signalr-client->ssi-fc-data==2.0.0) (1.14.3)
Requirement already satisfied: setuptools in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from gevent->signalr-client->ssi-fc-data==2.0.0) (41.2.0)
Requirement already satisfied: zope.interface in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from gevent->signalr-client->ssi-fc-data==2.0.0) (5.2.0)
Requirement already satisfied: zope.event in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (f
```

Bước 5: Giải nén và cài đặt client thành công:

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

Requirement already satisfied: greenlet>=0.4.17 in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from gevent->signalr-client->ssi-fc-data==2.0.0) (0.4.17)
Requirement already satisfied: cffi>=1.12.2 in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from gevent->signalr-client->ssi-fc-data==2.0.0) (1.14.3)
Requirement already satisfied: setuptools in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from gevent->signalr-client->ssi-fc-data==2.0.0) (41.2.0)
Requirement already satisfied: zope.interface in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from gevent->signalr-client->ssi-fc-data==2.0.0) (5.2.0)
Requirement already satisfied: zope.event in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from gevent->signalr-client->ssi-fc-data==2.0.0) (4.5.0)
Requirement already satisfied: pycparser in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from cffi>=1.12.2->gevent->signalr-client->ssi-fc-data==2.0.0) (2.20)
Requirement already satisfied: six in c:\users\anhnh2\appdata\local\programs\python\python38\lib\site-packages (from sseclient->signalr-client->ssi-fc-data==2.0.0) (1.15.0)
DEPRECATION: Source distribution is being reinstalled despite an installed package having the same name and version as the installed package. pip 21.1 will remove support for this functionality. A possible replacement is use --force-reinstall. You can find discussion regarding this at https://github.com/pypa/pip/issues/8711.
Using legacy 'setup.py install' for ssi-fc-data, since package 'wheel' is not installed.
Installing collected packages: ssi-fc-data
  Attempting uninstall: ssi-fc-data
    Found existing installation: ssi-fc-data 2.0.0
    Uninstalling ssi-fc-data-2.0.0:
      Successfully uninstalled ssi-fc-data-2.0.0
  Running setup.py install for ssi-fc-data ... done
Successfully installed ssi-fc-data-2.0.0

F:\tài liệu\PRO\FCdata\FC_PYTHON\FCData_PY>
```

### 2.1.2. 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 test\_Req\_Res.py.

- Chạy các API theo file test\_Req\_Res.py với câu lệnh: python <tên file muốn chạy>





```
1 auth_type = 'Bearer'
2
3
4 consumerID = '[redacted]'
5 consumerSecret = '[redacted]'
6
7 access_jwt = '[redacted]'
8
9
10 url = 'https://fc-data.ssi.com.vn/'
11 stream_url = 'https://fc-data.ssi.com.vn/'
```

- Lưu file config và Client đã được cài đặt thành công.

### 2.1.3. Market data

#### a. Streaming

Bước 1: Chạy file test\_Streaming.py bằng câu lệnh: `python test_Streaming.py`

Bước 2: Nhập kênh streaming là 1 trong các giá trị:

ALL: All channels

F: Securities Status

X: Market data

R: Foreign Room

MI: Index data

B: RealTime Bar (Ohlc Information)

VD như ảnh:



```
C:\Windows\System32\cmd.exe - python test_Streaming.py
^C
F:\tài liệu\PRO\FCdata\FC_PYTHON\FCData_PY>python test_Streaming.py
Please select channel: ALL
Streaming market data...
```

## b. API

1. Mở file test\_Req\_Res.py ( bằng Notepad ++...) và nhập thông tin cần kiểm tra
2. Nhập input cho hàm cần truy vấn từ số 11 -> 18

VD: Get SecuritiesDetails

Bước 1: Mở file test\_Req\_Res.py và nhập thông tin cần kiểm tra -> Save

```
config.py test_Req_Res.py
1 # import ssi_fc_trading
2 import ssi_fc_data
3 import config
4
5
6
7 def md_access_token():
8     print(ssi_fc_data.access_token(config))
9
10 def md_get_securities_list():
11     print(ssi_fc_data.securities(config, 'HNX', 1, 100))
12
13 def md_get_securities_details():
14     print(ssi_fc_data.securities_details(config, 'HNX', 'ACB', 1, 100))
15
16 def md_get_index_components():
17     print(ssi_fc_data.index_components(config, 'vn100', 1, 100))
18
19 def md_get_index_list():
20     print(ssi_fc_data.index_list(config, 'hnx', 1, 100))
21
```

Bước 2: Nhập số tương ứng với hàm SecuritiesDetails ở cmd -> Enter

C:\Windows\System32\cmd.exe - python test\_Req\_Res.py

F:\tài liệu\PRO\FCdata\FC\_PYTHON\FCData\_PY>python test\_Req\_Res.py

```
11 - Securities List
12 - Securities Details
13 - Index Components
14 - Index List
15 - Daily OHLC
16 - Intraday OHLC
17 - Daily index
18 - Stock price
19 - Get access token
Enter your choice: 11
```

Bước 3: Kết quả hiển thị như sau:

C:\Windows\System32\cmd.exe - python test\_Req\_Res.py

```
^C
F:\tài liệu\PRO\FCdata\FC_PYTHON\FCData_PY>python test_Req_Res.py
11 - Securities List
12 - Securities Details
13 - Index Components
14 - Index List
15 - Daily OHLC
16 - Intraday OHLC
17 - Daily index
18 - Stock price
19 - Get access token
Enter your choice: 11
{'data': [{'Market': 'HNX', 'Symbol': 'AAV', 'StockName': 'Công ty Cổ phần Việt Tiên Sơn Địa ốc', 'StockEnName': 'Viet Tien Son Real Estate Holding Company'}, {'Market': 'HNX', 'Symbol': 'ACM', 'StockName': 'Công ty Cổ phần Tập đoàn Khoáng sản Á Cường', 'StockEnName': 'A Cuong Mineral Group Joint Stock Company'}, {'Market': 'HNX', 'Symbol': 'ADC', 'StockName': 'Công ty Cổ phần Mỹ thuật và Truyền thông', 'StockEnName': 'Art Design and Communication Joint Stock Company'}, {'Market': 'HNX', 'Symbol': 'ALT', 'StockName': 'Công ty Cổ phần Văn hóa Tân Bình', 'StockEnName': 'Alta Company'}, {'Market': 'HNX', 'Symbol': 'AMC', 'StockName': 'Công ty Cổ phần Khoáng sản Á Châu', 'StockEnName': 'Asia Mineral Joint Stock Company'}, {'Market': 'HNX', 'Symbol': 'AME', 'StockName': 'Công ty cổ phần Alphanam E&C', 'StockEnName': 'Alphanam Mechanical Joint Stock Company'}, {'Market': 'HNX', 'Symbol': 'AMV', 'StockName': 'Công ty Cổ phần Sản xuất Kinh doanh Dược và Trang thiết bị Y tế Việt Mỹ', 'StockEnName': 'American Vietnamese Biotech Inc'}, {'Market': 'HNX', 'Symbol': 'API', 'StockName': 'Công ty Cổ phần Đầu tư Châu Á - Thái Bình Dương', 'StockEnName': 'Asia - Pacific Investment Joint Stock Company'}, {'Market': 'HNX', 'Symbol': 'APP', 'StockName': 'Công ty Cổ phần Phát triển Phụ gia và Sản phẩm dầu mỏ', 'StockEnName': 'Additives and Petroleum Products Joint Stock Company'}, {'Market': 'HNX', 'Symbol': 'APS', 'StockName': 'Công ty Cổ phần Chứng khoán Châu Á - Thái Bình Dương', 'StockEnName': 'Asia - Pacific Securities Joint Stock Company'}, {'Market': 'HNX', 'Symbol': 'ARM', 'StockName': 'Công ty Cổ phần Xuất nhập khẩu Hàng không', 'StockEnName': 'General Aviation Import Export Joint Stock Company'}, {'Market': 'HNX', 'Symbol': 'ART', 'StockName': 'Công ty Cổ phần Chứng khoán BOS', 'StockEnName': 'BOS Securities Corporation'}, {'Market': 'HNX', 'Symbol': 'ATS', 'StockName': 'Công ty Cổ phần Suất ăn công nghiệp Atesco', 'StockEnName': 'Atesco Industrial Catering Joint Stock Company'}, {'Market': 'HNX', 'Symbol': 'BAB', 'StockName': 'Ngân hàng Thương mại Cổ phần Bắc Á', 'StockEnName': 'Bac A Commercial Joint Stock Bank'}, {'Market': 'HNX',
```

## 2.2. NodeJS Client

### 2.2.1. Cài đặt

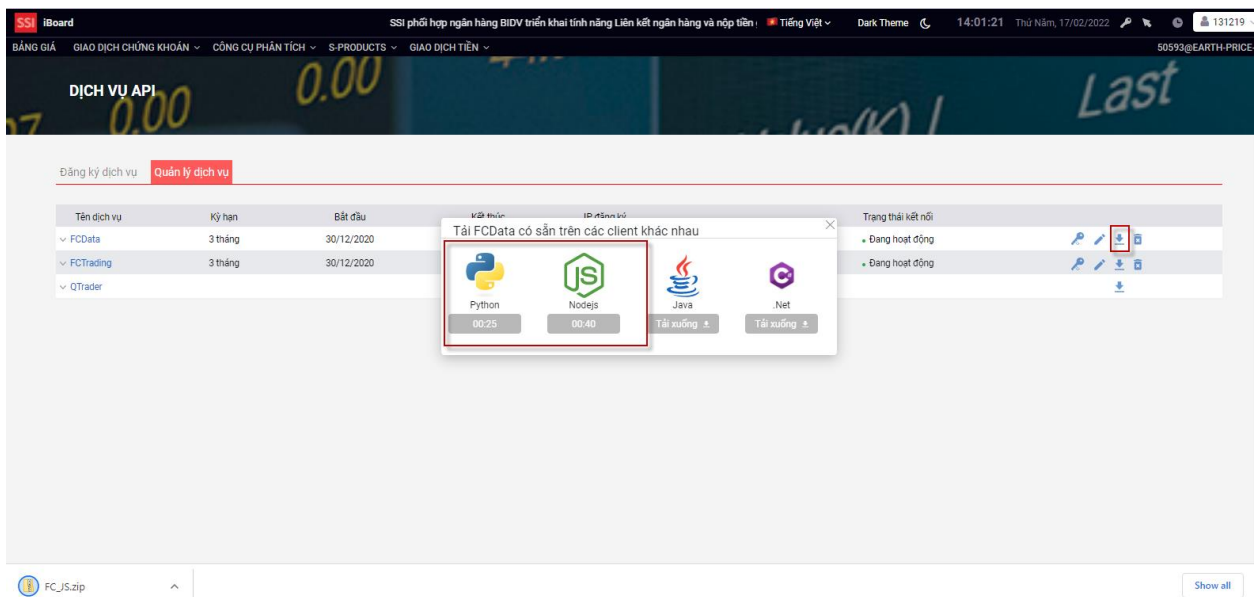
- Máy tính có cài đặt NodeJS ( cài đặt từ bản v.10.15.3 trở lên)
- Kiểm tra máy tính đã cài đặt NodeJS và version: cmd node –version

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

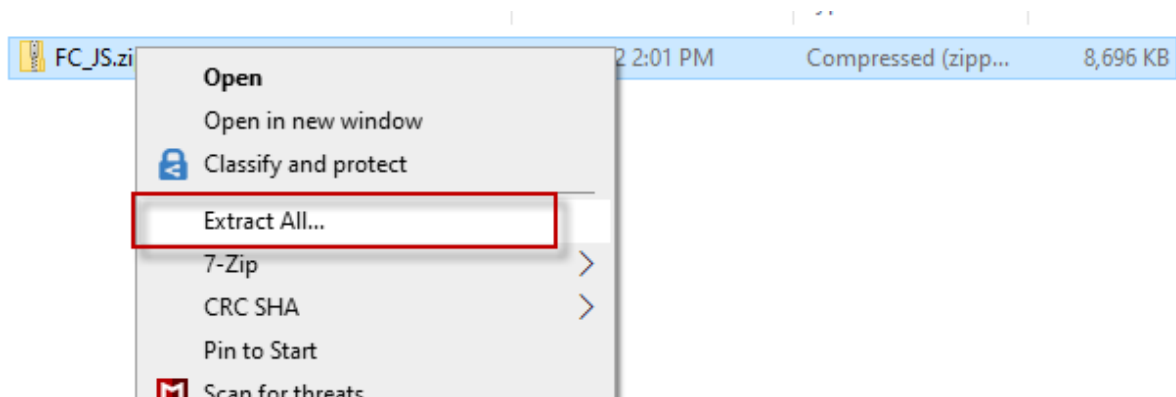
C:\WINDOWS\system32>cmd node --version
Microsoft Windows [Version 10.0.19043.1466]
(c) Microsoft Corporation. All rights reserved.

C:\WINDOWS\system32>
```

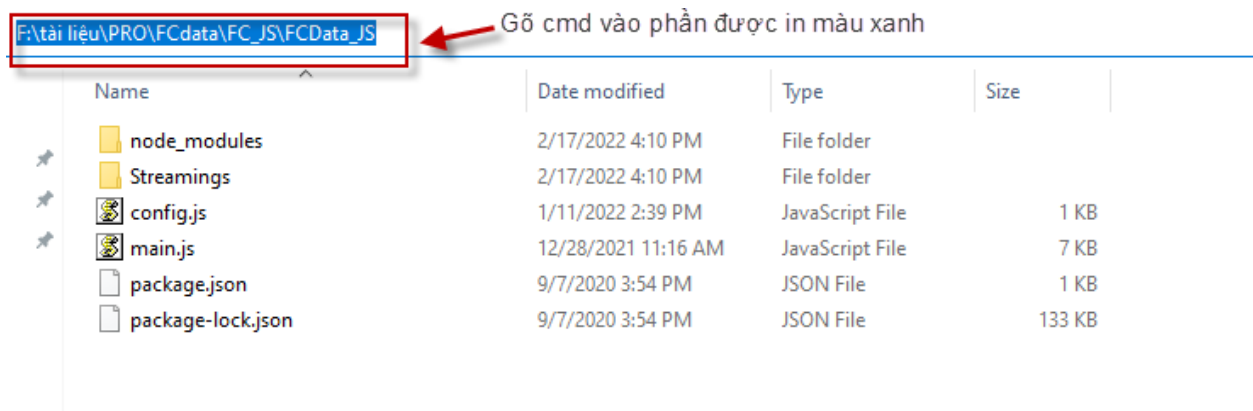
- Tải client và giải nén:



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\FCdata\FC_JS\FCData_JS>
```

Bước 3: Cài đặt Client theo câu lệnh sau: npm install -> Enter

```
C:\> npm
Microsoft Windows [Version 10.0.19043.1466]
(c) Microsoft Corporation. All rights reserved.
F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS>npm install
[.....] - postinstall: sill install executeActions
```

Bước 4: Khi chạy xong tiếp tục gõ câu lệnh: node main.js -> Enter

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

F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS>npm install
npm WARN ssi_fc_api@1.0.0 No description
npm WARN ssi_fc_api@1.0.0 No repository field.
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@1.2.9 (node_modules\fsevents):
npm WARN network SKIPPING OPTIONAL DEPENDENCY: request to https://registry.npmjs.org/fsevents/-/fsevents-1.2.9.tgz failed, reason: connect ETIMEDOUT 104.16.22.35:443

up to date in 273.396s

F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS>node main.js
Example app listening on port 3020!
Error: Request failed with status code 400
    at createError (F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS\node_modules\axios\lib\core\createError.js:16:15)
    at settle (F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS\node_modules\axios\lib\core\settle.js:18:12)
    at IncomingMessage.handleStreamEnd (F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS\node_modules\axios\lib\adapters\http.js:201:11)
    at IncomingMessage.emit (events.js:326:22)
    at endReadableNT (_stream_readable.js:1223:12)
    at processTicksAndRejections (internal/process/task_queues.js:84:21) {
  config: {
    adapter: [Function: httpAdapter],
    transformRequest: { '0': [Function: transformRequest] },
    transformResponse: { '0': [Function: transformResponse] },
    timeout: 5000,
    xsrfCookieName: 'XSRF-TOKEN',
    xsrfHeaderName: 'X-XSRF-TOKEN',
    maxContentLength: -1,
  },
}
```

Bước 5: Giải nén client thành công:

```
C:\Windows\System32\cmd.exe - node main.js
d, reason: connect ETIMEDOUT 104.16.16.35:443

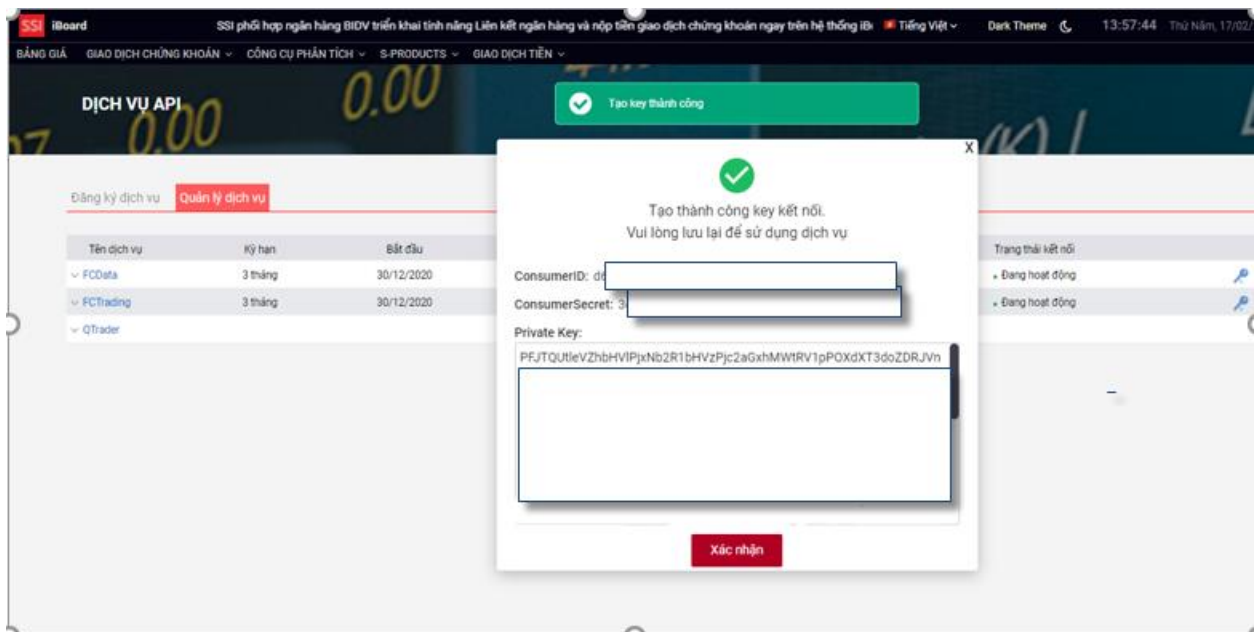
up to date in 277.481s

F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS>node main.js
Example app listening on port 3020!
Error: Request failed with status code 400
    at createError (F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS\node_modules\axios\lib\core\createError.js:16:15)
    at settle (F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS\node_modules\axios\lib\core\settle.js:18:12)
    at IncomingMessage.handleStreamEnd (F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS\node_modules\axios\lib\adapters\http.js:201:11)
    at IncomingMessage.emit (events.js:326:22)
    at endReadableNT (_stream_readable.js:1223:12)
    at processTicksAndRejections (internal/process/task_queues.js:84:21) {
  config: {
    adapter: [Function: httpAdapter],
    transformRequest: { '0': [Function: transformRequest] },
    transformResponse: { '0': [Function: transformResponse] },
    timeout: 5000,
    xsrfCookieName: 'XSRF-TOKEN',
    xsrfHeaderName: 'X-XSRF-TOKEN',
    maxContentLength: -1,
    validateStatus: [Function: validateStatus],
    headers: {
      Accept: 'application/json, text/plain, /*/*',
      'Content-Type': 'application/json;charset=utf-8',
      'User-Agent': 'axios/0.18.0',
      'Content-Length': 101
    }
  },
}
```

### 2.2.2. 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 main.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 config.js:



Trong đó:

+ id = ConsumerID

+ priKey = ConsumerSecret

➔ Lưu file config.js

Bước 3: Mở cmd và chạy lại câu lệnh: ctrl C -> node main.js

```
C:\Windows\System32\cmd.exe - node main.js
d, reason: connect ETIMEDOUT 104.16.16.35:443
up to date in 271.527s
F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS>node main.js
Example app listening on port 3020!
```

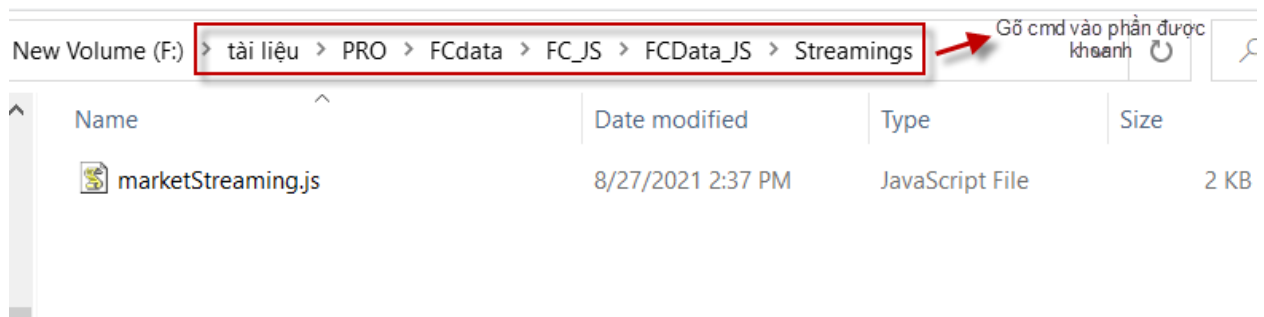
Bước 4: Giải nén và Cài đặt client thành công

```
C:\Windows\System32\cmd.exe - node main.js
d, reason: connect ETIMEDOUT 104.16.16.35:443
up to date in 271.527s
F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS>node main.js
Example app listening on port 3020!
```

### 2.2.3. Market data

#### a. Streaming

Bước 1: Gõ cmd vào ô địa chỉ thư mục để khởi tạo command line



Bước 2: Hiển thị màn hình cmd như ảnh:

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

F:\tai lieu\PRO\FCdata\FC_JS\FCData_JS\Streamings>
```

Bước 3: Vào file main.js xuống dòng 183 điền kênh muốn truy vấn -> Save



```
config.js x main.js x
165     consumerID: config.market.id,
166     consumerSecret: config.market.priKey,
167   }
168   }).then(response => {
169     if (response.data.status === 200) {
170       let token = "Bearer " + response.data.data.accessToken;
171       axios.interceptors.request.use(function (axios_config) {
172         axios_config.headers.Authorization = token;
173         return axios_config;
174       });
175
176       marketStreaming.initStream({
177         url: config.market.HubUrl,
178         token: token,
179       });
180
181       var mkClient = marketStreaming.start();
182
183       mkClient.serviceHandlers.connected = function (connection) {
184         mkClient.invoke(
185           'FcMarketDataV2Hub',
186           'SwitchChannels',
187           'X-QUOTE:ALL'
188         );
189       }
190
191       mkClient.serviceHandlers.reconnecting = function (connection) {
192         mkClient.invoke(
193           'FcMarketDataV2Hub',
194           'SwitchChannels',
195           'X:ALL'
196         );
197       }
198     } else {
```

Bước 4: Mở cmd và Điền câu lệnh: npm start

```
C:\> npm

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

F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS\Streamings>npm start

> ssi_fc_api@1.0.0 start F:\tài liệu\PRO\FCdata\FC_JS\FCData_JS
> node main.js
```

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 main.js

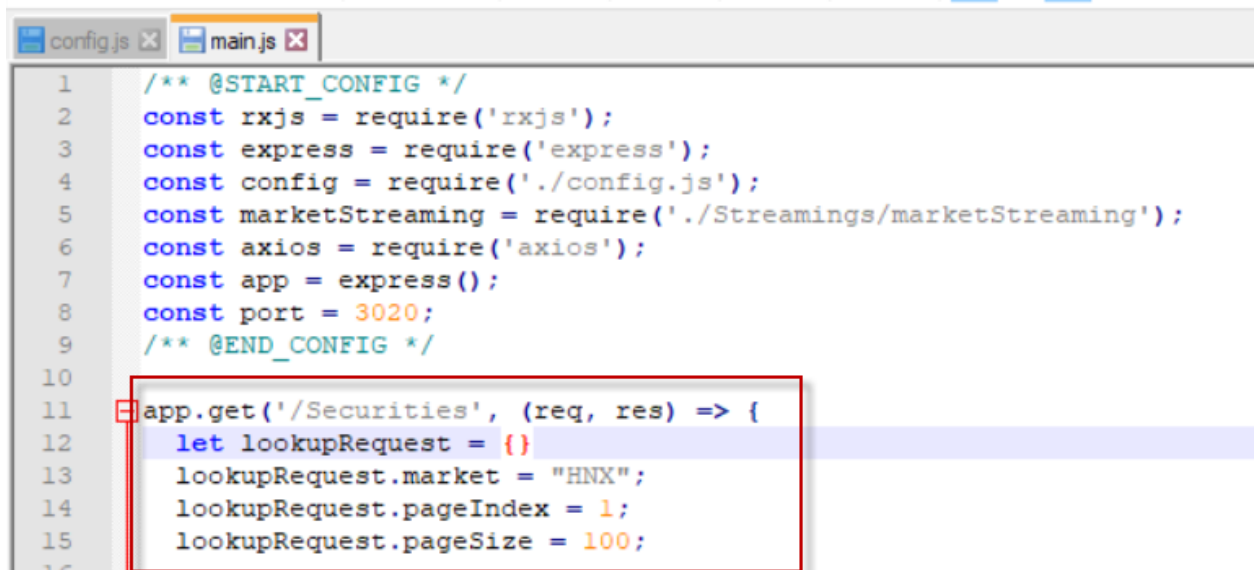
Bước 2: Lưu file và chạy lại cmd theo câu lệnh:

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

Bước 3: Kiểm tra thông tin vừa truy vấn

Ví dụ: Lấy thông tin Get Securities

- Điền thông tin muốn truy vấn ở hàm Securities trong file main.js -> Save



```
1  /** @START_CONFIG */
2  const rxjs = require('rxjs');
3  const express = require('express');
4  const config = require('./config.js');
5  const marketStreaming = require('./Streamings/marketStreaming');
6  const axios = require('axios');
7  const app = express();
8  const port = 3020;
9  /** @END_CONFIG */
10
11  app.get('/Securities', (req, res) => {
12    let lookupRequest = {};
13    lookupRequest.market = "HNX";
14    lookupRequest.pageIndex = 1;
15    lookupRequest.pageSize = 100;
```

- Mở trình duyệt và chạy câu lệnh: <http://localhost:3020/Securities>

← → ↻ ① localhost:3020/Securities

Apps học TA BA design crm job VOC SSI

```
{
  - data: [
    - {
      Market: "HOSE",
      Symbol: "LAF",
      StockName: "CTCP CHE BIEN HANG XUAT K",
      StockEnName: "Long An Food Processing Export Joint Stock Company"
    },
    - {
      Market: "HOSE",
      Symbol: "LBM",
      StockName: "CTCP VAT LIEU XD LAM DONG",
      StockEnName: "Lam Dong Minerals And Building Materials Joint Stock Company"
    },
    - {
      Market: "HOSE",
      Symbol: "LCG",
      StockName: "CT CP LICOGI 16",
      StockEnName: "LICOGI 16 Joint Stock Company"
    },
    - {
      Market: "HOSE",
      Svmbol: "LCM".
    }
  ]
}
```