**BÁO CÁO**

**Họ tên SV:** Nguyễn Quang Hưng

**MSSV:** 20183760

**Đề tài**: Crawl dữ liệu các mã cổ phiếu từ trang <https://banggia.hnx.vn> và xây dựng Web API lấy thông tin chi tiết của mã cố phiếu

**Giảng viên hướng dẫn**: Thầy Nguyễn Đức Tiến

# Tool Crawl dữ liệu cổ phiếu

## Công cụ Selenium và nguyên lý hoạt động

Selenium là bộ kiểm thử tự động miễn phí (mã nguồn mở) dành cho các ứng dụng web trên các trình duyệt và nền tảng khác nhau, tập trung vào việc tự động hoá các ứng dụng dựa trên nền tảng web. Bộ công cụ Selenium có 4 thành phần là Selenium Integrated Development Environment (IDE), Selenium Remote Control (RC), WebDriver, Selenium Grid trong đó WebDriver là đánh giá cao hơn cả. Selenium WebDriver tương thích với gần như tất cả các trình duyệt web phổ biến và hỗ trợ gần hết các ngôn ngữ lập trình phổ biến hiện nay. Trong selenium-crawl-hnx.py, em sử dụng công cụ Selenium WebDriver trên ngôn ngữ lập trình Python.

Nguyên lý hoạt động của Selenium WebDriver tạo một driver tới trình duyệt chạy trang banggia.hnx.để điều khiển trang web., thông qua tương tác trực tiếp với các trình duyệt. Để tương tác với các phần tử trên trình duyệt, cần tìm phần tử thông qua Xpath.

Để xác định Xpath các phần tử trên trang web, em dùng 1 extension là Selenium IDE. Extension này ghi lại tất cả các hành động thao tác với trang web và ghi lại địa chỉ của đối tượng vừa được thực thi. Ví dụ, em click vào 1 ô dữ liệu trên trang web thì tiện ích sẽ ghi lại sự kiện là onclick và địa chỉ tương ứng của nút bấm đó.

Sau đó khi đã có Xpath, gọi hàm **find\_element** để driver xác định phần tử rồi thực thi công việc lấy dữ liệu hoặc các sự kiện khác như click vào đối tượng.

Ví dụ:

driver.find\_element(By.XPATH, "//div[2]/div[2]/div/div/div")

## Phần crawl dữ liệu

File selenium-crawl-data.py

1. **Hàm crawl(url):**

Trên trang web banggia.hnx.vn, các mã cổ phiếu được chia làm 2 loại là cổ phiếu niêm yết (listed stock) và cổ phiếu chưa niêm yết (upcom stock). Mỗi trang cổ phiếu lại được chia thành nhiều section nhỏ hơn theo tên mã cổ phiếu, là các mục ‘ABC’, ‘DEF’, ‘GHI’, 'JKL', 'MNO', 'PQR', 'STUV', 'WXYZ'.

Hàm thực thi như sau:

* Tạo một driver điều khiển tới trình duyệt truy cập url = banggia.hnx.vn.
* Lần lượt gọi các sự kiện điều khiển click vào các mục cổ phiếu niêm yết và cổ phiếu chưa niêm yết.
* Với mỗi mục trên, sử dụng vòng lặp for gọi hàm crawl\_sections\_data để lấy dữ liệu từ các section nhỏ hơn (ABC, DEF, …) trên trang web.

Tham số đầu vào là url của trang web cần crawl.

Hàm trả về toàn bộ dữ liệu 1350 mã cổ phiếu đã crawl được.

1. **Hàm crawl\_sections\_data(driver, section):**

Tham số đầu vào là driver và tên section (như là ‘ABC’, ‘DEF’, ‘GHI’).

Hàm thực thi như sau:

* Driver điều khiển trình duyệt web click vào mục section.
* Sau đó gọi function lấy tất cả các hàng dữ liệu trong bảng dữ liệu được reload. Mỗi hàng chứa thông tin của một cổ phiếu bao gồm: mã cổ phiếu, giá tham chiếu, giá trần, giá sàn, giá bán, khối lượng bán, tổng khối lượng, tổng giá trị giao dịch, giá bán thấp nhất, giá bán cao nhất, giá bán trung bình.

Mỗi một cổ phiếu với các trường dữ liệu như thế sẽ được lưu vào 1 dictionary với các mục thông tin như sau:

* stock-code: mã cổ phiếu
* reference-price: giá tham chiếu
* celling-price: giá trần
* floor-price: giá sàn
* price: giá bán
* volume: khối lượng
* total-volume: tổng khối lượng
* total-value: tổng giá trị
* highest-price: giá bán cao nhất
* lowest-price: giá bán thấp nhất
* average-price: giá bán trung bình

Sau đó, các cổ phiếu được thêm vào một mảng là **stock\_list** là cấu trúc list chứa tất cả các thông tin của cổ phiếu

Hàm trả về list dữ liệu craw được

1. **Hàm export\_data(data)**

Hàm nhận tham số đầu vào là một cấu trúc list chứa dữ liệu của các mã cổ phiếu.

Hàm có tác dụng ghi các dữ liệu ra file .csv

1. **Hàm save\_database(data)**

Chức năng:

* Tạo một kết nối tới database bằng hàm getConnection() trong file connection\_utils.py.
* Hàm sẽ thực hiện 3 truy vấn là select, insert, update. Với mỗi mã cổ phiếu, tìm kiếm trên database xem đã tồn tại cố phiếu đó chưa (câu truy vấn select\_query). Nếu chưa tồn tại, thực hiện thêm mới vào (insert\_query). Nếu đã tồn tại, thực hiện cập nhật lại cố phiếu (update\_query).

1. **Hàm getConnection() trong file connection\_utils.py**

Chức năng: tạo mới một connection tới cơ sở dữ liệu mysql với các trường thông tin là địa chỉ host, username, password, tên database

## Sinh documentation

Chức năng sinh documentation cho file crawl-stock-hnx.py. Các hàm trong file đã được đặt comment theo quy tắc. File tài liệu sinh ra dưới dạng html, xem trự tiếp trên web

Lệnh chạy trên terminal: *python -m pydoc -w .\selenium-crawl-hnx.py*

## Log error và gửi mail

Chức năng ghi log được sử dụng khi phần mềm gặp phải lỗi (error). Em đặt các đoạn try-catch. Khi xảy ra lỗi, các Exception được bắt và phần mềm sẽ tự động ghi log chi tiết lỗi ra file log-crawl-hnx cùng với gửi mail về địa chỉ email đã được cài đặt trước.

Ở đây, bản thân công cụ selenium đã có chức năng ghi log ở level DEBUG. Vậy em đã tắt ghi log ở cấp độ DEBUG và thực hiện ghi log.error

# Xây dựng Web API

## Công cụ sử dụng

Em sử dụng **nodejs** để xây dựng server Web API. Sau đó đẩy server lên 1 trang web miễn phí là **heroku.com.**

Server cung cấp một api ở địa chỉ:

*http://vnindex.herokuapp.com/getStockInfo?code={code}*

với code là mã cố phiếu do người dùng nhập vào

## API getStockInfo

API sẽ nhận tham số đầu vào là mã cổ phiếu do người dùng nhập. Sau đó tạo một kết nối đến cơ sở dữ liệu vnindex để truy vấn.

API trả về một json với cấu trúc gồm các trường:

* stock-code: mã cổ phiếu
* reference-price: giá tham chiếu
* celling-price: giá trần
* floor-price: giá sàn
* price: giá bán
* volume: khối lượng
* total-volume: tổng khối lượng
* total-value: tổng giá trị
* highest-price: giá bán cao nhất
* lowest-price: giá bán thấp nhất
* average-price: giá bán trung bình
* created-at: ngày khởi tạo
* updated-at: ngày cập nhật

Ví dụ:

Application

Description automatically generated with medium confidence

Nếu không tìm thấy cố phiếu đó trên cơ sở dữ liệu, API sẽ trả về là {mã cổ phiếu} not found

Ví dụ: Graphical user interface, text, application

Description automatically generated