BÁO CÁO BÀI TẬP 01

Họ tên: Huỳnh Minh Huấn

MSSV: 1612858

1. Basic Info Scraping

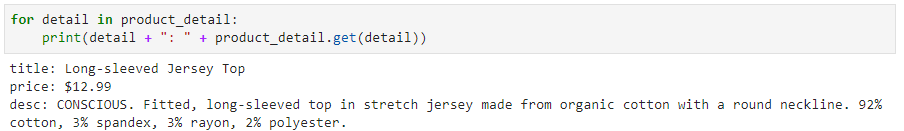
* Em sử dụng thư viện requests-html.
* Gọi lớp HTMLSession, để get url của website <https://scrapingclub.com/exercise/detail_basic/>.
* Dùng CSS selector ‘div.card-body’, với tham số first = True-> lấy ra content về card-body đầu tiên.



* Trong card\_body, sử dụng hàm find tương tự ở trên để tìm thông tin về title, description và price theo các thẻ con trong <div.card-body>

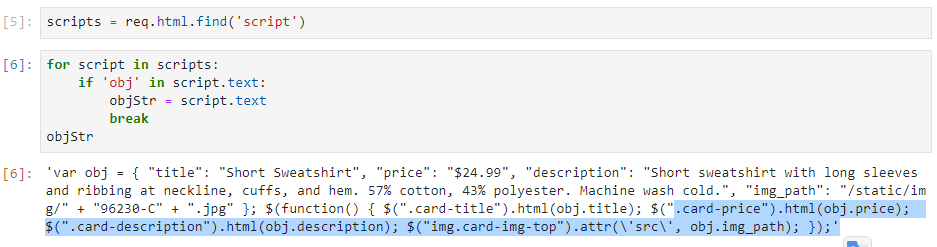


* Rút được thông tin về sản phẩm

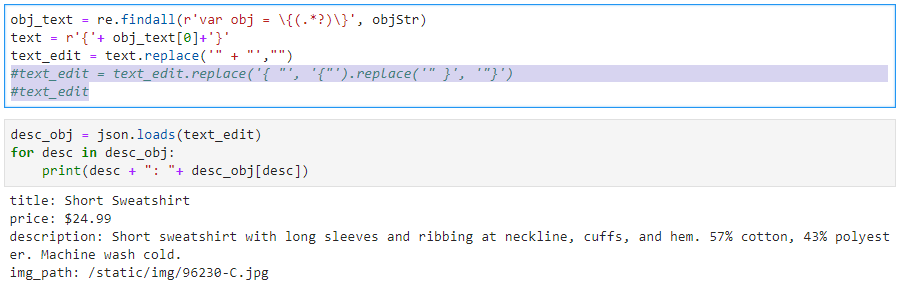


1. Analyze JSON

* Em sử dụng thư viện requests-html, json và re (regex).
* Gọi lớp HTMLSession, để get url của website <https://scrapingclub.com/exercise/detail_json/>.
* Dùng hàm find để filter content theo script, sau đó xử lý chuỗi để lấy được chuỗi “var obj = { …”



* Sử dụng regex + json để lấy được nội dung obj

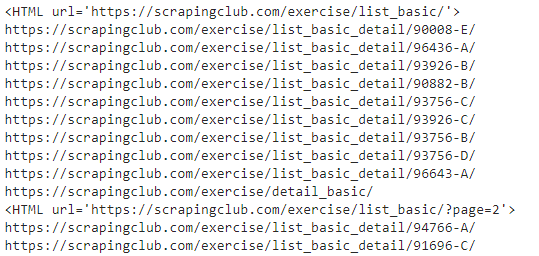


1. Recursively Scraping pages

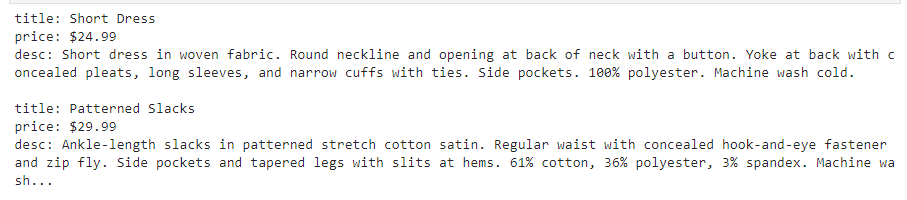
* Em sử dụng thư viện requests-html, time.
* Gọi lớp HTMLSession, để get url của website <https://scrapingclub.com/exercise/list_basic>.
* Requests-html hỗ trợ phương thức .next() để lấy link kế tiếp phục vụ cho pagination (link về pagination: <https://requests-html.kennethreitz.org/#pagination>.).
* Hàm get\_sub\_page để lấy info của các trang con khi get vào các trang đó.



* Với mỗi card tại url chính sẽ có 1 link đến trang con, em gọi hàm trên để lấy thông tin ở trang con đó.



* Kết quả: một vài kết quả

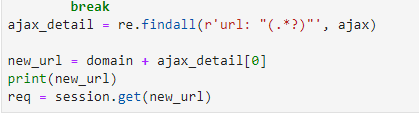


1. Mimicking Ajax requests

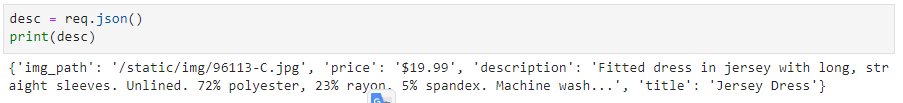
* Mục tiêu bài này là extract được link ajax\_detail (<https://scrapingclub.com/exercise/ajaxdetail/>) để truy cập vào get object json
* url gốc là <https://scrapingclub.com/exercise/detail_ajax/>.
* Em sử dụng thư viện requests-html, re và json.
* Dùng hàm phương thức find() của HTMLSession để tìm script chứa link cần extract



* Sử dụng regex để extract link

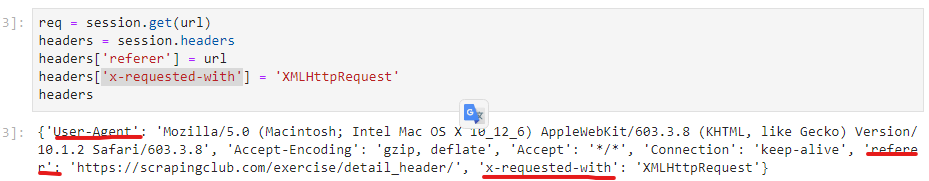


* Get obj json bằng phương thức .json()

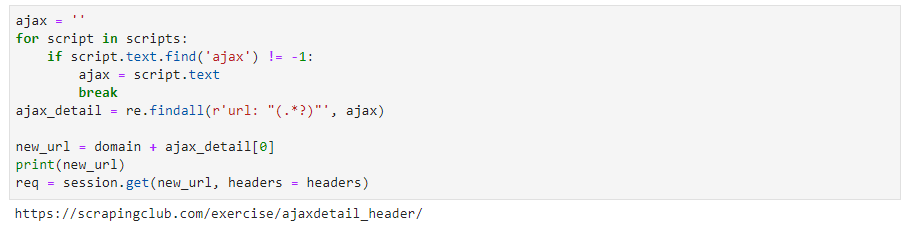


1. Inspect HTTP request

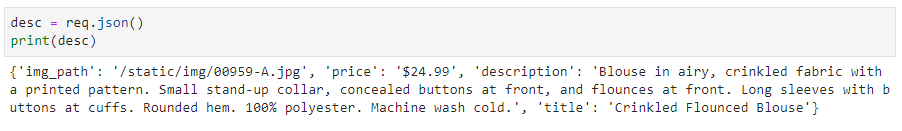
* Tương tự exercise 04.
* Cần thêm headers (referer và 'x-requested-with') mới có thể get được nội dung link extract từ ajax script <https://scrapingclub.com/exercise/ajaxdetail_header/>



* url gốc <https://scrapingclub.com/exercise/detail_header/>
* extract link tương tự exercise 04, phương thức .find() filter theo tag script + regex



* kết quả:



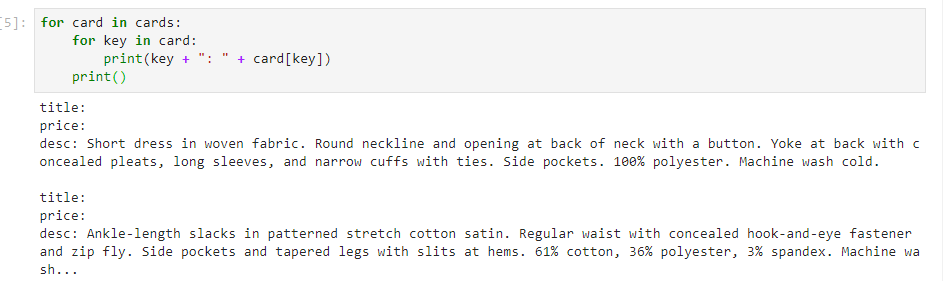
1. Scraping Infinite Scrolling Pages (Ajax)

* Em sử dụng thư viện requests-html và time.
* Tương tự câu 3, việc scrolling (pagination) được requests-html hỗ trợ.
* Hàm get\_sub\_page để lấy thông tin sản phẩm của card.



* Filter theo thẻ card bằng phương thức .find để lấy ra các card -> extract link chi tiết.



* Kết quả
* 

1. Find gold in cookie

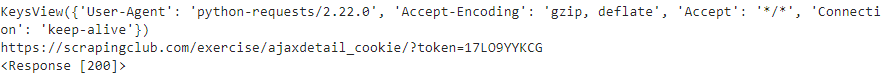
* Có thể sử dụng thư viện requests hay requests-html, thư viện bs4 để tiện lấy content, thư viện re (regex).
* Cần lấy headers và token. Token được extract từ headers (response) ‘set-cookie’.

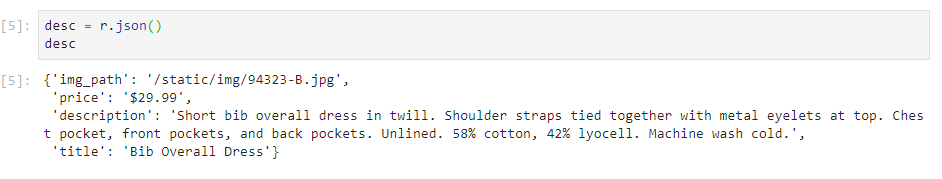
 : get url để nhận được response -> extract token

: cần headers với 2 nội dung này ‘user-agent’ và ‘x-requested-with’.

: extract token -> tạo link url để get

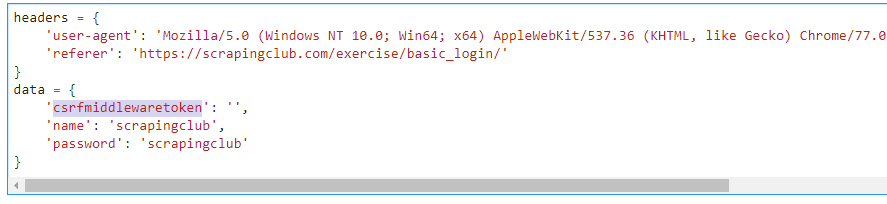
*  : get url để nhận response -> chứa content thông tin hàng.



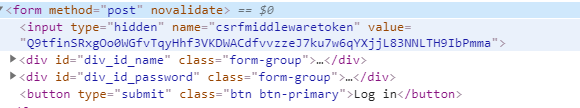
* Kết quả:
* 

1. Login form

* Em sử dụng thư viện requests và bs4
* Để điền form đăng nhập, ta cần headers của requests và data.



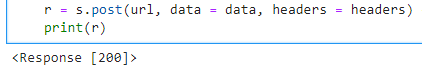
* Trong data thì ‘csrfmiddlewaretoken’ sẽ unique với mỗi session

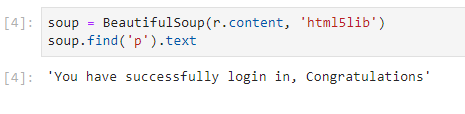


* Gọi phương thức get của session trong requests để lấy gái trị ‘csrfmiddlewaretoken’ trong session đó



* Gọi phương thức post của session để send data form thực hiện log in



* Kết quả:
* 

1. Solve Captcha

* Tương tự exercise 08, cần thêm 2 nội dung truyền vào data form để send post lên là captcha\_0 và captcha\_1

1. Decode minified javascript
2. Cách 1: Sử dụng Selenium để load website



1. Cách 2:

Tham Khảo:

Documentation:

* Requests: <https://requests.kennethreitz.org/en/master/user/quickstart/#custom-headers>.
* Requests-html: <https://requests-html.kennethreitz.org/>.

Books:

* <https://marcell.memoryoftheworld.org/Richard%20Lawson/Web%20Scraping%20With%20Python%20(2685)/Web%20Scraping%20With%20Python%20-%20Richard%20Lawson.pdf>.