PAUTROT
Aurélien
ID : K-13388

UNIVERSITY OF WARSAW

Webscraping project

Doctolib

PAUTROT
Aurélien
ID : K-13388

UNIVERSITY OF WARSAW

Description of the project

Doctolib.fr is a French website regrouping most of French doctors and medical centers from all specialities. The aim of the project is to gather useful information about the doctors. I chose to work on the page about dentists in Paris. The base page of consist in a list of 10 doctors and medical centers that leads to their personal page with all information by clicking on it. Anyway, all the pages have the same pattern so the scrapers I wrote can be used for every medical specialisations and places in France by simply changing the base URL.

Description of scrapers mechanics

BautifulSoup : The BeautifulSoup scraper start by grabbing links to doctors page from the basd page (10 doctors link per page) on the defined number of pages to scrape (100 by default). Then, it goes from doctors link to link to gather data for every single doctor or medical center.

Selenium : The Selenium scraper start on the base page. It click on the link of the first doctor and open it in a new tab (to avoid some errors when reloading the base page). It gather data from the doctor page, add it to the database then close the tab, go back to the base page and do the same thing for each doctor on the page. When every doctors page of the base page have been scraped, the scraper go to the next page by clicking on the "next" button and restart the same operations until it scraped the number of pages predefined (100 by default).

Scrapy : The Scrapy scraper is divided in two spiders. The first spider is dedicated to gathering links to doctors pages from base page. It gather links for all pages in the limit of the predefined number (100 by default). The spider must be run with the command "scrapy crawl spider1 -o links.csv" to store the links used by the second spider. The second spider grab links from the links.csv previously created with the first spider and use them to scrape every single doctor or medical center pages and output the data when runed with "scrapy crawl spider2 -o data.csv"

Description of the outputs

Output	Description
Name	Return the name of the doctor / medical center
Number	Return the phone number of the doctor / medical center
Price	Return the minimum price for an appointment with the doctor
Address	Return the address of the doctor / medical center
Zip_code	Return the zip code, use to know the district of the doctor / medical center
Vital_card	Return "Carte vitale acceptée" if the doctor accept the vital card, which means the appointment is partially or totally refunded by the French social security

PAUTROT
Aurélien
ID : K-13388

UNIVERSITY OF WARSAW

Data analysis

District	Prix	% vital card	Index
Paris 1thr District	23,13	67%	35
Paris 2th District	35,24	50%	70
Paris 3th District	28,80	86%	34
Paris 4th District	26,43	64%	42
Paris 5th District	41,63	53%	79
Paris 6th District	36,08	85%	43
Paris 7th District	33,19	73%	46
Paris 8th District	39,85	67%	59
Paris 9th District	30,34	61%	50
Paris 10th District	23,17	59%	39
Paris 11th District	23,38	47%	50
Paris 12th District	23,30	55%	42
Paris 13th District	23,19	52%	44
Paris 14th District	26,19	69%	38
Paris 15th District	23,90	65%	37
Paris 16th District	44,69	68%	65
Paris 17th District	34,00	57%	59
Paris 18th District	23,00	42%	55
Paris 19th District	23,09	39%	59
Paris 20th District	40,79	57%	72
Total général	**30,17**	**60%**	**50**

The first analysis we could make with the database we made is a comparison of prices depending on the district. The average lower price for a basic appointment is **30,17 €**. However, based on our sample, the repartition of price seems very dependant of the District. The price of an appointment in the 16th district is for example nearly 2 times more expensive than in the 18th.

The percentage of doctors accepting the vital card, which leads to a refund of the fees, is also very inequal depending on the place.

I used these 2 data to build a qualitative index that show the best and worst places to get an appointment in Paris, according to the minimum price and percentage of accepting the vital card (price * 1/vital card %). The higher the index is, the worst it is, and vice versa. These data could be joined with data of other medical specialists (generalists, Ophthalmologist, Dermatologist, gynecologist, etc.) scraped by my scrapers and then create an index that would show the global medical landscape of each District of Paris, or even other city of France.

The data could also be useful to medical professionals to check the best places to work, etc.

PAUTROT
Aurélien
ID : K-13388

UNIVERSITY OF WARSAW

Performance analysis

BeautifulSoup	Scrapy	Selenium
33 minutes	**3** minutes **24** seconds	**100** minutes

Scrapy is undeniably the faster, by a lot. This is mainly due to the fact that it don't need to wait every action to be executed to start the followings. BeautifulSoup and Selenium are slower. For my case, selenium is way slower, but this is due to certain points :

- First, I needed to put huge time.sleep (5 seconds each) because my connection was not fast enough to load the pages otherwise.
- Second, the way I parsed the site. The fact that I need to click on every doctor link then go back etc. make loose a lot of time. If the website had all information on the base page without the need to follow links, Selenium would have performed almost the same speed as BeautifulSoup.

Selenium is not worth to use here since there is no need to connect, send keys, etc.