Numbeo scraping

Authors:

Klaudia Bury (396188) – selenium

Maria Janiszewska (396783) – beautiful soup

Bartłomiej Bollin (396175) – scrapy

Short description: Numbeo is a website created in 2009 by Mladen Adamovic. It gathers data for a few categories (Cost of living, Clime, Climate etc.). In this research we wanted to focus on a Cost of living in each city over the world which can be an interesting field to explore.

Scraper mechanics: To scrap data from the website we created 3 scrapers each using different method, which are presented below.

Scrapy:

The project consists of 3 scrapers called countries, cities and prices. The first one scrapes just a single page and gathers the list of all available countries. The second one takes its output and crawls the webpage of each country to get the list of all cities for each one. Finally, the prices scraper takes this output and crawls every single city page and extracts the data about prices in different categories. The entire project is accompanied by a Python3 script that calls spiders in the required order and provides a command line interface to customize the output size and format. To avoid clutter the scrapy log level is set to INFO and is redirected to a scrapy.log file. All spiders use a custom pipeline defined in pipelines.py so that the columns are in a certain order (although changing it won't break the process as the csvs are read to a pandas.DataFrame and the column order is ensured at this level as well). All items used in the project are defined in items.py.

Beautiful soup:

The script is divided in 3 parts: 1. Get names of the country - where countries are extracted from the dropdown list. 2. Extract links to cities in each country - where for each country the program will find cities (from dropdown list) and prepare links for them. In here you can find a limit condition. 3. Get data for each city - as soon as the links are prepared program gets data for selected variables: country, city, category, price, minimum and maximum price.

Selenium:

The script starts from gathering links to all countries on the homepage. In the next step, it visits each country page where it scrapes names of cities from the drop down lists. At that point Selenium doesn't directly click on each city, but uses the following GET Request instead: https://www.numbeo.com/cost-of-living/city_result.jsp?country={}&city={}&displayCurrency=USD where it puts country and city name respectively inside curly brackets. Later it gets redirected to each city website separately, and scrapes all data about items and their prices. Afterall, it is all put into a collective dataframe, which is then exported to the csv file at the very end.

Expected output:

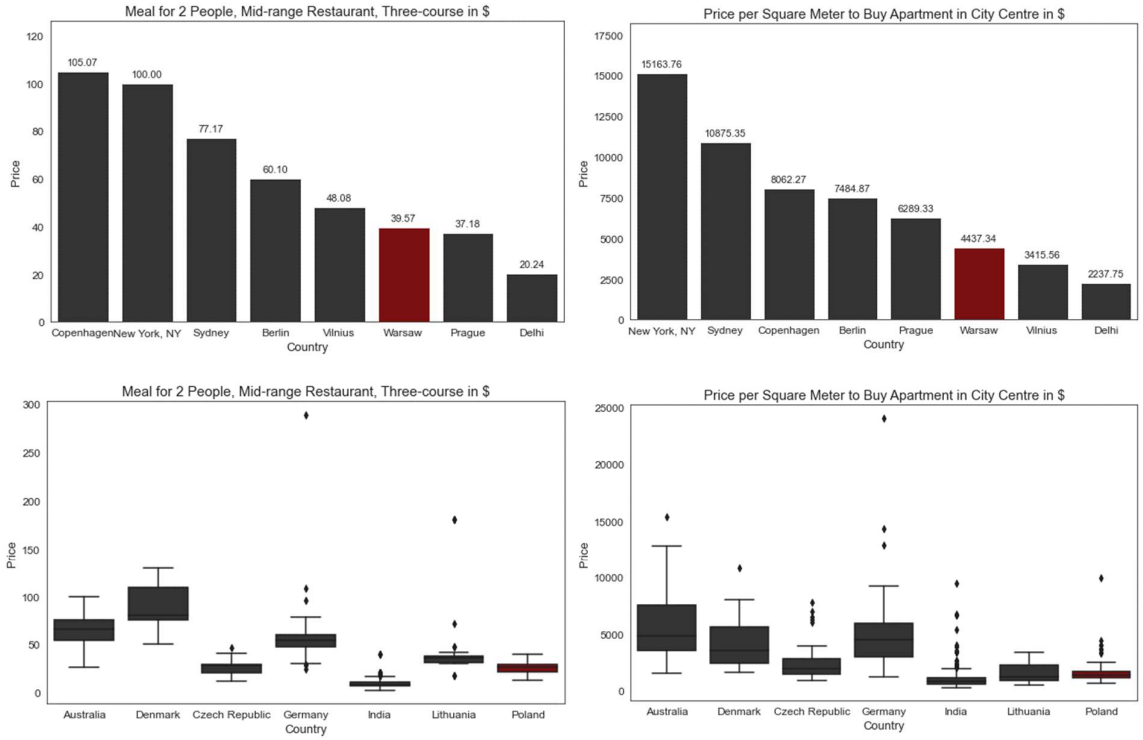
After running each scraper we get a data frame with columns: Country, City, Category, Name, Price, Min price, Max which we can easily export to csv. Prices are scraped in USD. Please note that data may differ each time you scrape, due to ongoing changes on the website.

Comparison of scrapers

	Scrapy	Beautiful soup	Selenium
time for first 1000 pages	62 s	2840 s	6489 s

In order to compare three scrapers we run each of them with the limit of 1000 pages. As it turned out *scrapy* is the most efficient tool to deal with many pages. The time that it took to scrape the data was about one minute long so it is not even comparable with the rest. *Selenium* scraper is the slowest one as expected so using it to scrape big amount of data from the static websites is not a good idea. However it is a very important tool to deal with dynamic websites where interaction is needed, for example to test functionality of a website.

Data analysis:



The above analysis shows, that scraped data allows us to statistically compare price levels across cities/countries. For instance, one can see how price levels for restaurant meals and apartments in big cities compare to the capital city of Poland – Warsaw (marked in red).