Report web app

For my code, I divided it in 4 parts for the back part. There is the part that scrape the Michelin website, the part that scrape the MaitreRestaurateur website and the third one index.js that makes the link between the two previous one. I stored all the collected data in two different json files, one for each site respectively.

Thanks to axios and cheerios, I was able to scrape each site and with the devtool of my favorite browser I was able to find the directory/link to scrape. Then, I built two different scrapers for the two sites and I was able to return clean and exploitable data.

*michelin.js:*

With the function parseRestaurant, I collect all the url from the pages of bib filter in the presentation of the restaurants. Once all thoses url returned to the get function, I scrape every restaurant associated with this url, then I parse the collected data and store it into the bibList.json file. With a console.table, we can observe our data in console, I put one at the good spot to show it.

*maitreRestaurateur.js:*

With the function httpGet, I make a, http request to an address that I found in the sources of the maitrerestaurateur website. With a POST request I collect data for each page of the website that I increment. Once the data collected, I parsed them in order to have clean data and stored them into the MrList.json file. We can also easily visualize the data with the console.table that I before the end. Surprisingly, my code that originally was able to scrape all the pages, finally decided to scrape every time the first page, even if I’m increasing the page number. With more time, I would be able to investigate more on that.

The next step is to compare the data from the two json and to combine them into one. As my maitreRestaurateur.js has flaws I can’t use it to combine otherwise we would be limited. I’ll show only the first 50 from bib Gourmand.