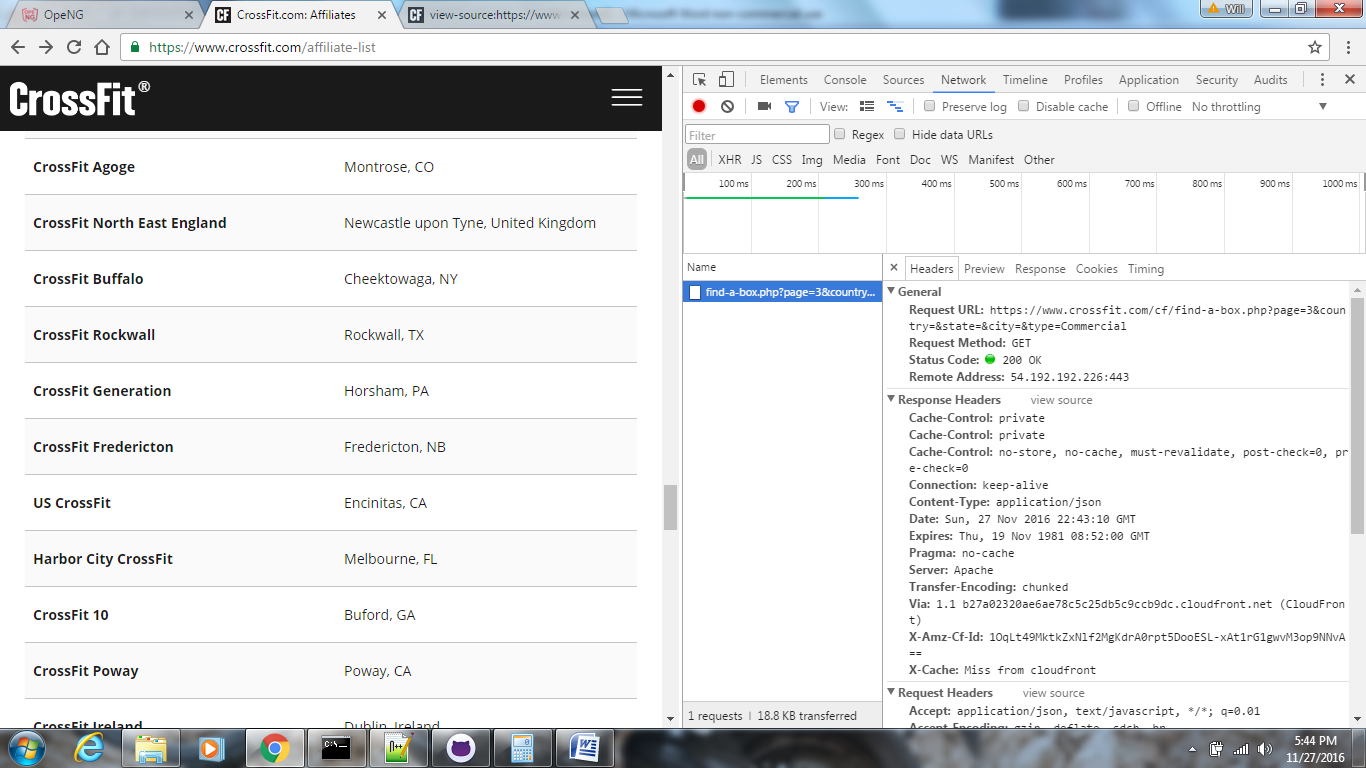
Thanks to the API at openg.azurewebsites.net, the tables containing the event score data is readily available in JSON format and easily ingested by Python requests module. Likewise, athlete profile data on games.crossfit.com is static and easily scrappable using BeautifulSoup. I ran into an issue with the CrossFit Affiliates listing (https://www.crossfit.com/affiliate-list) in that it uses an AJAX loader to load additional gyms once the browser scrolls to the bottom of the page. This means that this additional data isn’t available in the HTML file and readable by BeautifulSoup. But I know it is somewhere nearby. At some point during loading, the page has to request this data. Therefore, I went to the handy-dandy Chrome Developer Tools.

By going to the Network tab and reloading the page, we can view each of the GET calls made during loading and one of them MUST be out table loader. Due to the nature of the AJAX loader, simply scroll to the bottom of the page and see what loads on the Networks developer table.



I now know that the page is providing parameters to a PHP file (<https://www.crossfit.com/cf/find-a-box.php?page=3&country=&state=&city=&type=Commercial>) to load the data. By following the link, you’ll see that the PHP file is returning JSON formatted data. I already see a way that I could simply loop through multiple calls to this PHP file supplying increasing page numbers until the returned JSON object = {“affiliates”:[]}. How cool is that? I actually found a website attempting to sell this data set that we can now pull on our own for $74. Forget that.