


August 2022

WEB SCRAPING PROJECT

NBA TOP 100 PLAYER



Prepared by: Nijat Gurbanov (444424)

OUTLINE

- Introduction
- Beautiful Soup
- Selenium
- Scrapy
- Conclusion

Introduction

The main aim of this project is to scrape some basic information about the TOP 100 NBA players based on PPG (Points Per Game) from the [RealGM website](#).

The RealGM website is a website that shows statistics, news, and information about basketball players.

My project consists of three sections. Each section uses different tools to scrape the same information. At the end of each section, the

	Name	Current Team	Age	Nationality	Height (in cm)	Weight (in kg)
0	Joel Embiid	Philadelphia Sixers	28	Cameroon	213	127
1	Nikola Jokic	Denver Nuggets	27	Serbia	213	129
2	Ja Morant	Memphis Grizzlies	23	United States	191	79
3	Jayson Tatum	Boston Celtics	24	United States	203	95
4	Trae Young	Atlanta Hawks	23	United States	185	82
..
95	Luke Kennard	Los Angeles Clippers	26	United States	196	93
96	Devonte' Graham	New Orleans Pelicans	27	United States	185	84
97	Jae'sean Tate	Houston Rockets	26	United States	193	104
98	Alec Burks	Detroit Pistons	31	United States	198	97
99	Danilo Gallinari	Boston Celtics	34	Italy	208	106

Beautiful Soup

The section of Beautiful Soup consists of three steps. First, it scrapes the links of the profile of the TOP 100 NBA players. Then from each of scraped links, our program extracts the needed information of each player. At the end, our program make an extremely simple analysis on extracted data:

```
Simple data analysis  
Average age: 26.88  
Average height: 198.76 cm  
Average weight: 96.74 kg  
The team with most strongest players is Boston Celtics
```

Total running time of the program was 80.33 seconds.

Selenium

The methadology of Selenium is similar to "human way". If we extract those data ourselves, then we would enter each profile, get the data from there and go back to the main site and repeat the process. Selenium does the same steps itself. It also extracts the same information and make the same analysis. However, Selenium works extremely slowly in this case. It depends on user's internet connection, however in avarage it takes approximately 10-13 minutes to scrape the pages.

Scrapy

I used two spiders to extract the needed information from the webpage. The first spider (links) extracts the links of the player profiles. The second one (players) uses those links and extracts the needed information from the players' page. You have to save the results as a CSV file manually. For running spiders, you have to use the following codes in the UBUNTU WSL terminal.

- `$ scrapy crawl links -O links.csv`
- `$ scrapy crawl players -O players.csv`

Moreover, there is also one additional file (simple-analysis.py) which makes a simple data analysis over data. The result is the same with others. The most beautiful part of Scrapy is the speed. Scrapy took less than 10 seconds to run the whole program (both spiders) which is much better than the other two methods.

Conclusion

In a conclusion, I used three different methods to scrape the same information. The scrapy method works much more faster than other two methods.

