Web Scrapping of a Car Details Website

# Introduction:

This is a Assignment task given by Pfactorial. The main theme of this task is scrape the website and collect the details from that website([www.gotoauto.ca](http://www.gotoauto.ca)). This is a car details website. We want to take car price, car name and car description from inventory section of this website. I do this task by using AutoScraper. Then I make csv file of car price and car names also I make text files of 5 cars details.

# About:

This is a car details website. I scrape the car details from this website by using AutoScraper .and make csv file of car price and car names also make text files of car details. I make 5 cars details text files

# AutoScraper:

AutoScraper is the web scraping python library that is used for scraping data from a website in a simple, easy and fast way. It has a user-friendly environment by this scraper can easily interact with this library. It uses the URLs and HTML content of the website for scraping reliable information and data. It learns scraping rules and returns similar elements in a good format.

# Steps and Tasks:

* Firs I install AutoScraper by using ” pip install autoscraper”
* Then I import the Autoscraper library by using “ from autoscraper import AutoScraper”
* Also I import pandas for making Dataframes “Import pandas as pd”
* Then I started my task 1

# Task 1:

* Take Car name and Car price(MSRP) from Inventory.
* For this I take (<https://www.gotoauto.ca/inventory/>) , I scrapped this url and collect car price and car name.
* For wanted list of scrapping I take car price and car names. Because these are the details we want to scrape.
* After scraping I groped car names and car price by using “scrapper.get\_result\_similar(auto\_url,grouped=True)”
* After that I used same url. because when we use similar url. we can get car names and car price very fastly. code: results=scrapper.get\_result\_similar("https://www.gotoauto.ca/inventory/",group\_by\_alias=True)
* Then I make a dictionary. I make a dictionary of car names and car prices. code: “Data={'Car Name':results['Title'], 'Car Price':results['Price']}”
* I convert my dictionary to Data Frame. Because I can take csv file of that dictionary. code: “DataFrame=pd.DataFrame(Data)”
* Then I take csv file of that Data Frame. The name of csv file is 'task\_1.csv.'

# Task 2:

* In task2 the main theme is collect the details of car.
* For this I used this (<https://www.gotoauto.ca/inventory/view/14862358/)url>
* For this I take car description as the wanted list
* Then I take 5 car details by using this code: results\_details=scrapper.get\_result\_similar("https://www.gotoauto.ca/inventory/view/14936812/",group\_by\_alias=True)
* Then I make 5 dictionary, each dictionary is for each car details. code: result\_details\_data={'Description':results\_details['Description']}

results\_details2\_data={'Description':results\_details2['Description']}

results\_details3\_data={'Description':results\_details3['Description']}

results\_details4\_data={'Description':results\_details4['Description']}

results\_details5\_data={'Description':results\_details5['Description']}

* Then I convert each dictionary to each data frame. Because I can take text file very easily from data frame.

result\_details\_dataframe=pd.DataFrame(result\_details\_data)

result\_details2\_dataframe=pd.DataFrame(results\_details2\_data)

result\_details3\_dataframe=pd.DataFrame(results\_details3\_data)

result\_details4\_dataframe=pd.DataFrame(results\_details4\_data)

results\_details5\_data={'Description':results\_details5['Description']}

* After this I take text file of each car description dataframe.

# Conclusion:

I conclude this task. The AutoScraper is a powerful web scraping python library. That is used for scraping data from a website in a simple, easy and fast way. So this technology is very good. I can scrape every details like car price, car name and car details in easy way. I make csv file of car price and car name, also I make text files of 5 cars details.