Web Scraping Car Details from Cars24.com

Project Report(Team-M)

*Objective* - The main objective of this mini-project is to develop skills in web scraping by extracting and analysing car details of **MG HECTOR CARS** from Cars24.com, focusing on different locations. The key details to be gathered include kilo meters driven, year of manufacture, fuel type, transmission, and price. This mini-project is important as it provides hands-on experience in web scraping, a crucial skill for data extraction from online sources. By focusing on different locations, it also allows for comparative analysis, revealing regional differences in car attributes and pricing. Ultimately, the project enhances data cleaning and structuring skills, essential for any data analysis task.

*Methodology* -- To achieve the objective of extracting and analysing car details from Cars24. com we started by setting up the environment, installing necessary Python libraries such as **requests**, **beautifulsoup4**, and **pandas**. The next step involved sending HTTP requests using the requests library to retrieve the HTML content of web pages containing car listings from Cars24.com. Once the HTML content was obtained, we utilized Beautiful Soup to parse the content and locate the elements containing relevant car details. These extracted details were then stored in a structured format using Pandas. After compiling the data, we cleaned and formatted it to ensure consistency and readability. The final output is a well-structured dataset that provides insights of the car markets in Mumbai, Pune, Gurgaon, Hyderabad, Ghaziabad, Chennai, Bengaluru and Noida focusing on the specified attributes.

#### Tools and Libraries –

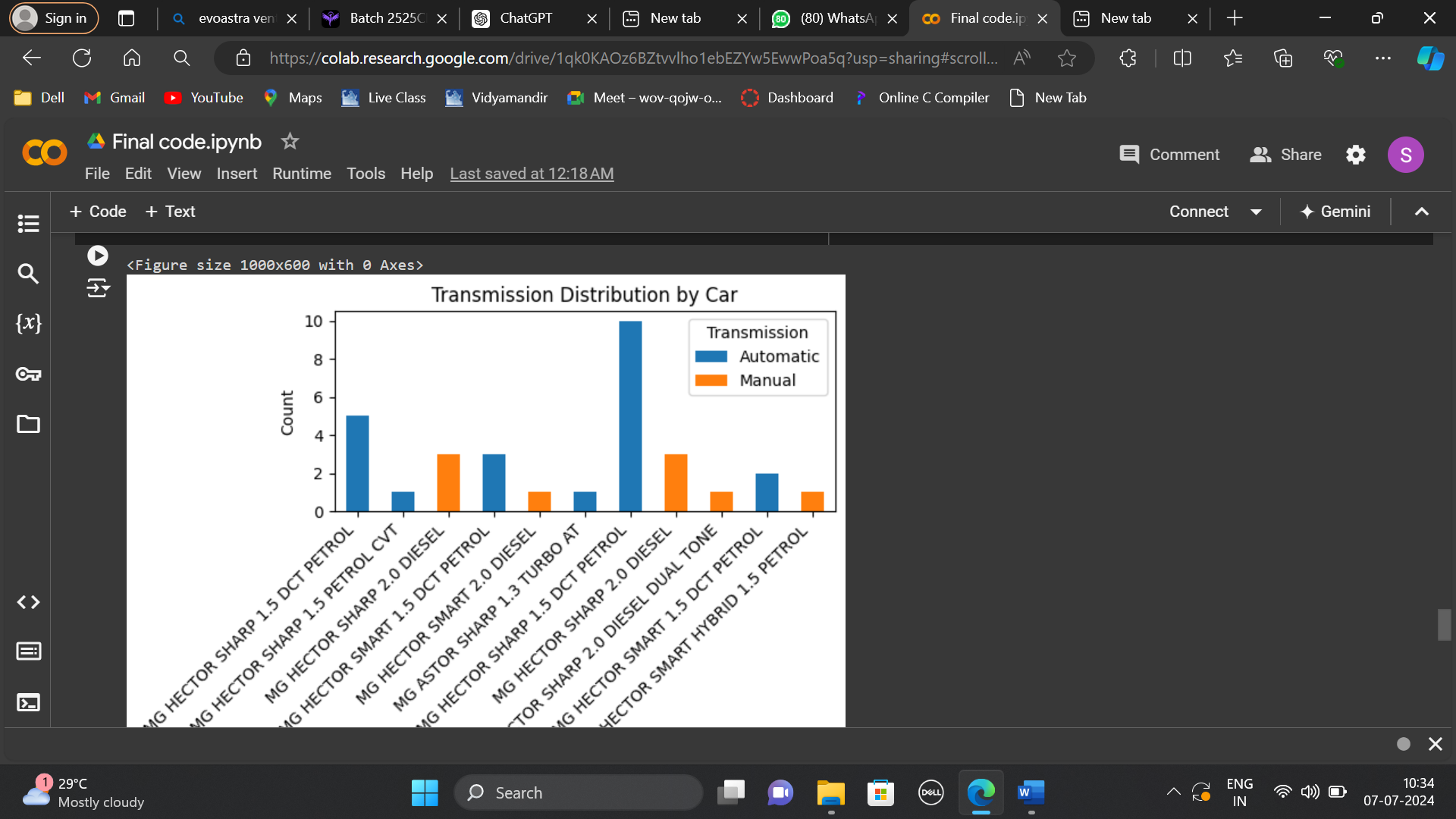
* **Python**: Programming language used for scripting.
* **Beautiful Soup**: Library for parsing HTML and XML documents.
* **Requests**: Library for making HTTP requests.
* **Pandas**: Library for data manipulation and analysis.
* **Google Collab**: Environment for running Python code and documenting the process.
* **CSV**: Module for reading and writing CSV (Comma Separated Values) files.

*Result Analysis-*

The analysis shows that the average kilo meters driven for cars in the dataset is around the mean value, indicating typical usage levels, with an average price of ₹14.65 lakh for used cars in different locations.

The stacked bar chart illustrating transmission distribution by car model reveals significant insights into the prevalence of manual versus automatic transmissions. This visualization helps to identify models with a higher count of either transmission type, highlighting consumer preferences. These findings provide valuable benchmarks for potential buyers, aiding in assessing car usage, market pricing, and transmission trends in the Mumbai used car market.

The average kilo meters driven for cars in the dataset is 35,535 km. This relatively moderate mileage suggests that the cars listed are neither heavily used nor brand new. This information is valuable for potential buyers as it indicates that the vehicles have a reasonable amount of wear and tear, which can affect their performance and maintenance needs.



The 2022 MG HECTOR SMART HYBRID 1.5 PETROL, has a high mileage of 127,904 km, indicating extensive use which could affect its condition and future maintenance despite its recent manufacture. Priced at ₹18.08 lakh, its value might be high considering the extensive usage, although its hybrid technology and new model status could justify the cost.

The 2019 MG HECTOR SHARP 1.5 DCT PETROL, features very low mileage at 3,306 km, suggesting minimal use and making it an attractive option. Priced at ₹12.86 lakh, it offers a diesel engine and automatic transmission, making it a good value for those seeking a lightly used, efficient, and convenient vehicle.

*Challenges Faced-*

**Combining Datasets from Multiple Locations into One Dataset**: Integrating datasets from multiple locations poses challenges in standardizing data formats, handling varying column names or structures, and reconciling overlapping or conflicting data entries. Each location dataset may have unique identifiers or naming conventions, requiring preprocessing steps such as renaming columns, standardizing data formats, and resolving data conflicts to create a unified dataset.

**Ensuring Accuracy and Completeness of Data**: Maintaining data accuracy and completeness is critical in web scraping projects, particularly when aggregating information from the websites. Challenges include variations in data formats, missing or inconsistent data fields, and occasional inaccuracies in scraped information.

**Handling Complex HTML Structures** : When using Beautiful Soup to scrape data from websites, as it heavily depends on specific patterns in the website's HTML to find and collect information. If the HTML structure is messy or irregular—like having unclear tags or nested elements—it can make it hard for Beautiful Soup to locate all the data correctly. This situation might cause errors or miss out on some data during scraping. To handle this, you might need to carefully study the website's HTML layout and possibly adjust your scraping code to handle these complexities better. It's also essential to build in safety measures to catch and manage any errors that might occur, ensuring that the data you scrape is complete and accurate.