

TABLE OF CONTENTS

1. INTRODUCTION -

- 1.1 Objective.
- 1.2 What is Data Scraping
- 1.3 Scope

2. BACKGROUND AND LITERATURE SURVEY -

- 2.1 Software Requirement Specifications.

3. DESIGN -

- 3.1 Data Flow Diagram.
- 3.2 Use Case Diagram.
- 3.3 Activity Diagram.

4. TECHNOLOGY USED -

- 4.1 Python.
 - 4.1.1 Beautiful Soup.
 - 4.1.2 Request.
 - 4.1.3 Pandas.
 - 4.1.4 Writer.
- 4.2 Excel.
- 4.3 Power BI.

5. CODING AND IMPLEMENTATION -

6. FUTURE ENHANCEMENT -

7. CONCLUSION -

8. BIBLIOGRAPHY -

LIST OF FIGURES

Figure No.	Title
1.	A basic Data Scraping Diagram.
2.	Data Flow Diagram.
3.	Use Case Diagram.
4.	Activity Diagram.

CHAPTER : 1

INTRODUCTION

1.1 Objective

Main Objective of Data Scraping is to extract information from one or many websites and process it into simple structures such as spreadsheets, database or CSV file. The goal of this thesis is to build a basic program to extract a data set with some specific information provided by the clients according to their needs. The concept of scraping the web is not new, however, with modern programming languages it is possible to build web scrapers that can collect unstructured data and save this in a structured way.

1.2 What is Data Scraping

Data scraping, in its most general form, refers to a technique in which a computer program extracts data from output generated from another program. Data scraping is commonly manifest in web scraping, the process of using an application to extract valuable information from a website.

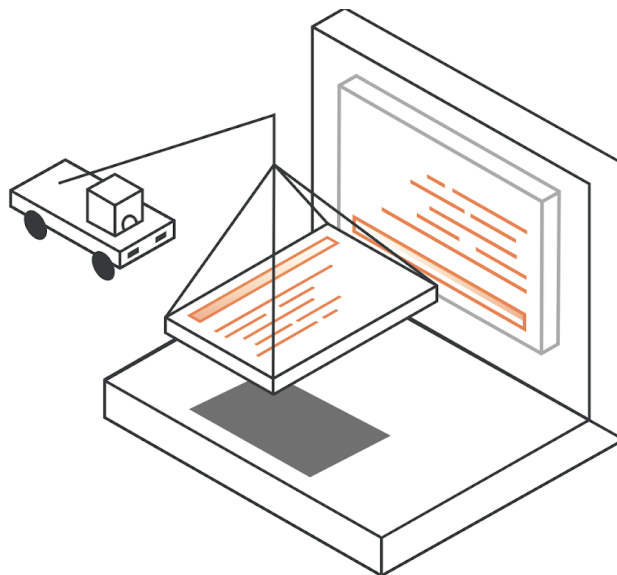


Figure 1.1 Data scraping

Why scrape website data?

Typically companies do not want their unique content to be downloaded and reused for unauthorized purposes. As a result, they don't expose all data via a consumable API or other easily accessible resource. Scraper bots, on the other hand, are interested in getting website data regardless of any attempt at limiting access. As a result, a cat-and-mouse game exists between web scraping bots and various content protection strategies, with each trying to outmaneuver the other.

The process of web scraping is fairly simple, though the implementation can be complex. Web scraping occurs in 3 steps:

1. First the piece of code used to pull the information, which we call a scraper bot, sends an HTTP GET request to a specific website.
2. When the website responds, the scraper parses the HTML document for a specific pattern of data.
3. Once the data is extracted, it is converted into whatever specific format the scraper bot's author designed.

Scraper bots can be designed for many purposes, such as:

1. Content Scraping - content can be pulled from the website in order to site in order to replicate the unique advantage of a particular product or service that relies on content. For example, a product like Yelp relies on reviews; a competitor could scrape all the review content from Yelp and reproduce the content on their own site, pretending the content is original.
2. Price scraping - by scraping pricing data, competitors are able to aggregate information about their competition. This can allow them to formulate a unique advantage.
3. Contact scraping - a lot of websites contain email addresses and phone numbers in plaintext. By scraping locations like an online employee directory, a scraper is able to aggregate contact details for bulk mailing lists, robo calls, or malicious social engineering attempts. This is one of the primary methods both spammers and scammers use to find new targets.

How is web scraping mitigated?

Typically, all content a website visitor is able to see must be transferred onto the visitor's machine, and any information a visitor is able to access can be scraped by a bot. Efforts can be made to limit the amount of web scraping that can occur. Here are 3 methods of limiting exposure to data scraping efforts:

1. Rate limit requests - for a human visitor clicking through a series of webpages on a website, the speed of interaction with the website is fairly predictable; you'll never have a human browsing 100 webpages a second, for example. Computers, on the other hand, can make requests orders of magnitude faster than a human, and novice data scrapers may use unthrottled scraping techniques to attempt to scrape an entire website very quickly. By rate limiting the maximum number of requests a particular IP address is able to make over a given window of time, websites are able to protect themselves from exploitative requests and limit the amount of data scraping that can occur within a certain window.
2. Modify HTML markup at regular intervals - data scraping bots rely on consistent formatting in order to effectively traverse website content and parse out and save useful data. One method of interrupting this workflow is to regularly change elements of the HTML markup so that consistent scraping becomes more complicated. By nesting HTML elements, or changing other aspects of the markup, simple data scraping efforts will be hindered or thwarted. For some websites, each time a webpage is rendered, some form of content protection modifications are randomized and implemented. Other websites will change up their markup code occasionally to prevent longer-term data scraping efforts.
3. Use CAPTCHAs for high-volume requesters - in addition to using a rate limiting solution, another useful step in slowing content scrapers is the requirement that a website visitor answers a challenge that's difficult for a computer to surmount. While a human can reasonably answer the challenge, a headless browser* engaging in data scraping most likely

cannot, and certainly will not consistently across many instances of the challenge. However, constant CAPTCHA challenges can negatively impact the user experience

Another less common method of mitigation calls for embedding content inside media objects like images. Because the content does not exist in a string of characters, copying the content is far more complex, requiring optical character recognition (OCR) to pull the data from an image file. But this can also hinder web users who need to copy content such as an address or phone number off a website instead of memorizing or retyping it.

*A headless browser is a type of web browser, much like Chrome or Firefox, but it doesn't have a visual user interface by default, allowing it to move much faster than a typical web browser. By essentially running at the level of a command line, a headless browser is able to avoid rendering entire web applications.

How is web scraping stopped completely?

The only way to totally stop web scraping is to avoid putting content on a website entirely. However, using an advanced bot management solution can help websites eliminate access for scraper bots almost completely.

What is the difference between data scraping and data crawling?

Crawling refers to the process large search engines like Google undertake when they send their robot crawlers, such as Googlebot, out into the network to index Internet content. Scraping, on the other hand, is typically structured specifically to extract data from a particular website.

1. Scraper bots will pretend to be web browsers, while a crawler bot will indicate its purpose and not attempt to trick a website into thinking it's something it is not.
2. Sometimes scrapers will take advanced actions like filling out forms, or otherwise engaging in behaviors to reach a certain part of the website. Crawlers will not.
3. Scrapers typically have no regard for the robots.txt file, which is a text file containing information specifically designed to tell web crawlers what data to parse and what areas of the

site to avoid. Because a scraper is designed to pull specific content, it may be designed to pull content explicitly marked to be ignored.

1.3 Scope

This project will only focus on building the platform needed to reach the goals. No hardware will be developed and only CarTrade.com will be scraped. The website will only do what it needs to reach the goals.

Handling scraped data outside of the goals of the project will not be a part of the programming, and the web scrapper will only be able to handle CarTrade. the website will only be able to handle said data from that specific webscraper.

CHAPTER : 2

BACKGROUND AND LITERATURE SURVEY

2.0 Software Specification and Literature Survey

Requirement specifications is a way of summarizing the requirements that the customer and developer has on the product or program. It is often done in different types of lists and stretches from design to functions and backend. When all the demands in the requirement specification is met the product is done.

This report and project can be broken into a few different steps with the over-arching goal of building a platform where users can web scrape their Cartrade pages, upload this parsed, structured, and cleansed data a website where the data gets stored in a database. This data can then be presented in a visually pleasing way to the user. The different processes chosen for the project are presented together with the main steps done to ensure that the project is followed through.

2.1 Time planning

Due to the scope of the project the structure and planning are key elements to achieve the set goals. Three different objects need to be built with a python-based web scraper, a dynamic website that can handle active users and a database to store both log-in information regarding the user, but also the scraped data that the user uploads. To achieve this a Gant-based schedule was chosen to plan how the time was divided between the three different objects and writing the report.

2.2 Development process

A development process was created early in the project to go from a concept and idea to developed product. The development process for this project can be divided into 6 different steps.

2.2.1 Pre-study and data collection

The first step in the pre study is to analyze what needs to be done. How much research is needed to learn how to build a python-based web scraper. During this step questions like if code from previous projects can be re-used or not is answered. It's also the step where the baseline of the platform is outlined. Developing a Python based web scraper – A study on the development of a web scraper for Cartrade.

2.2.2 Requirement specification and user stories

The second step is to do a requirement specification. What must be developed to accomplish the project based on the goals and boundaries that has been previously mentioned. Creating a solid requirement specification is important so the development has an end goal. The requirement is usually what the project needs to accomplish when done.

2.2.3 Functional analysis

The third step is to do a functional analysis. Step two, requirement specification, makes it clear what the projects requirements are, what it needs to achieve, while functional specification is a continuation of this. The functional analysis is done to evaluate what functions the platform needs to have. These functions are then presented in a table that clearly outlines what the developed product needs to be able to do.

2.2.4 Traditional development

During the project iterative and agile development was used. A lot of knowledge was gained, and a lot of research was read during the project, which mean that the most important part of development was having an iterative pipeline where code and functions could be changed and iterative as the development continued. The result of the development is the finished project.

2.2.5 Python

Python is a well-developed strong language with a vast library, with the most important one for this project being BeautifulSoup for web scraping. Different languages were considered for this project the choice of Python was motivated through previous experience with the language and the existence of the web scraping library BeautifulSoup.

2.2.6 BeautifulSoup

Beautiful Soup is the library used for this project because of all the well documented functions. There are other libraries that are as easy to use, such as lxml, but BeautifulSoup has a big and dedicated user base that publishes solutions to a lot of problems. This makes it easy to troubleshoot issues on the internet and reach out for help. For more advanced projects or other types of web scraping better libraries exist, however, the learning curve of these can be steep and too big for the scope of this project.

2.3 Evaluation

Once everything has been developed an evaluation is done to see if the new system is better than the old system. This is a way of quality control to make sure that the goals of the project were reached. The evaluation will be done measuring two different variables. The number of clicks needed for both the new system and the old system, and the amount of time needed for the old system and the new system. This will then be presented under chapter 4. Developing a Python based web scraper – A study on the development of a web scraper for CarTrade. The number of clicks is measured through counting the number of clicks from start to result. The amount of time is measured by the amount of time from start to presented result. The tests are done by the author of this report, and done back to back to ensure the fairest amount of measurements.

2.4 Presentation

When the development process is done, and the coding and development of the project is finished a technical report is written. This report includes the theory behind the project, the tools used and how the project was constructed. During the entire project an iterative design and development process was used to make sure that time was allocated as best it could be and that the project reached completion in time for the deadline.

CHAPTER : 3

DESIGN

3.1 Data Flow Diagram

It's easy to understand the flow of data through systems with the right data flow diagram.

What is Data Flow Diagram?

A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles and arrows, plus short text labels, to show data inputs, outputs, storage points and the routes between each destination. Data flowcharts can range from simple, even hand-drawn process overviews, to in-depth, multi-level DFDs that dig progressively deeper into how the data is handled. They can be used to analyze an existing system or model a new one. Like all the best diagrams and charts, a DFD can often visually “say” things that would be hard to explain in words, and they work for both technical and nontechnical audiences, from developer to CEO. That's why DFDs remain so popular after all these years. While they work well for data flow software and systems, they are less applicable nowadays to visualizing interactive, real-time or database-oriented software or systems.

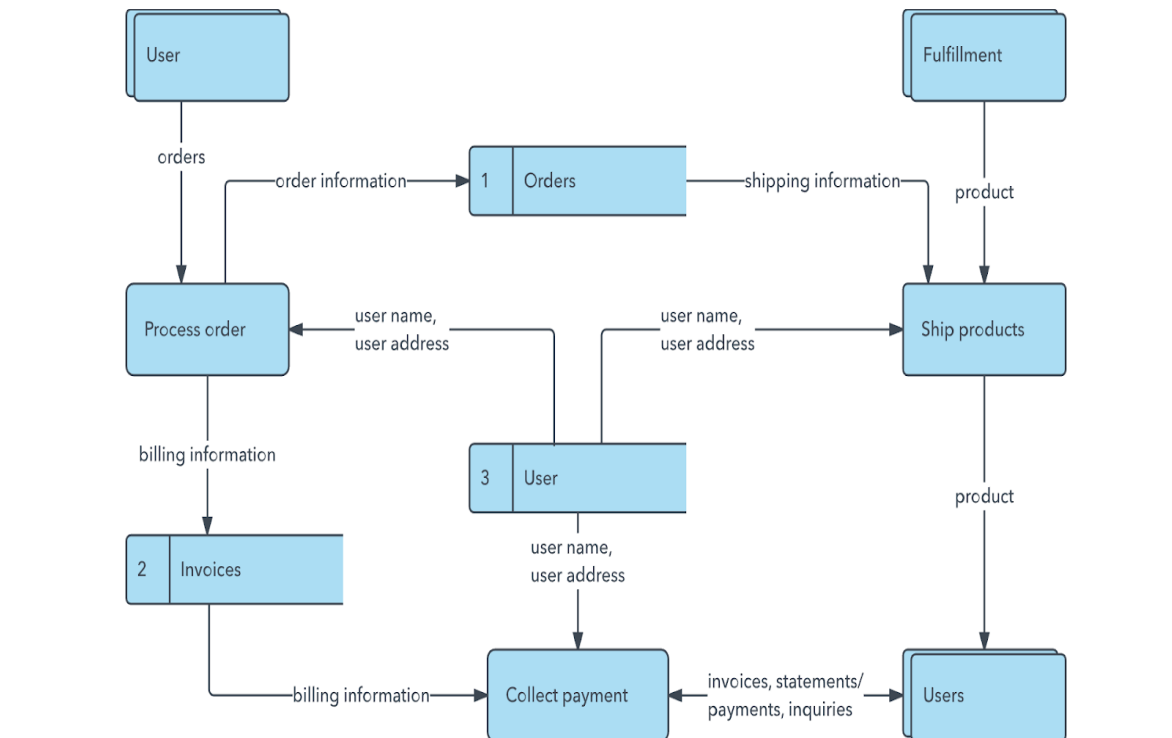


Figure 3.1 Data flow overview diagram

History of Data Flow Diagram

Data flow diagrams were popularized in the late 1970s, arising from the book structural design, by computing pioneers Ed Yourdon and Larry Constantine. They based it on the “data flow graph” computation models by David Martin and Gerald Estrin. The structured design concept took off in the software engineering field, and the DFD method took off with it. It became more popular in business circles, as it was applied to business analysis, than in academic circles.

Data Scraping Data Flow Diagram -

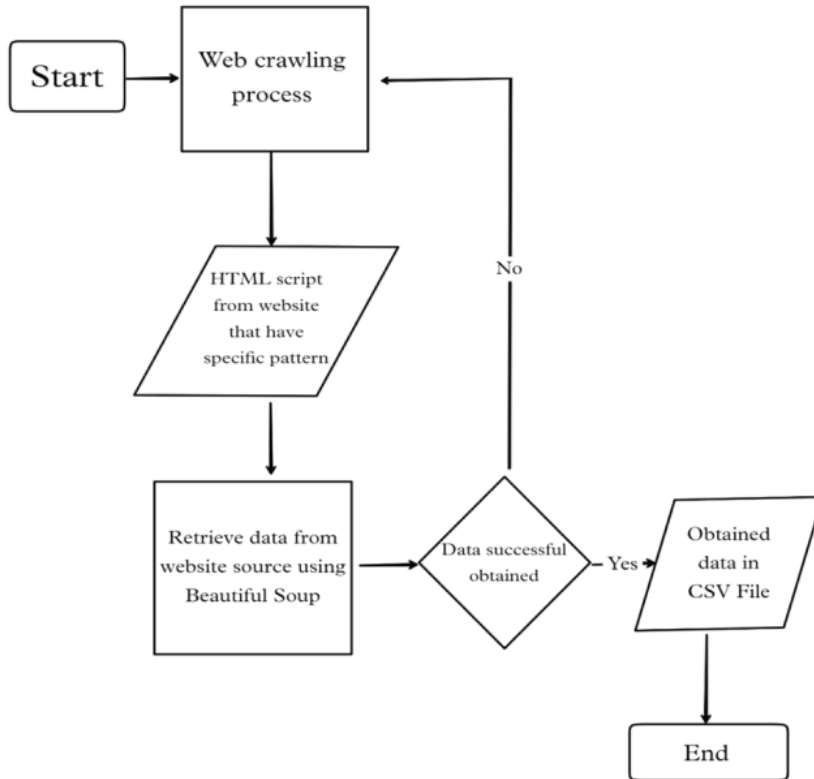


Figure 3.2 Data flow diagram

3.2 Use Case Diagram

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and also tells how the user handles a system.

Purpose of Use Case Diagrams

The main purpose of a use case diagram is to portray the dynamic aspect of a system. It accumulates the system's requirement, which includes both internal as well as external influences. It invokes persons, use cases, and several things that invoke the actors and elements accountable for the implementation of use case diagrams. It represents how an entity from the external environment can interact with a part of the system.

Following are the purposes of a use case diagram given below:

1. It gathers the system's needs.
2. It depicts the external view of the system.
3. It recognizes the internal as well as external factors that influence the system.
4. It represents the interaction between the actors.

How to draw a Use Case Diagram?

It is essential to analyze the whole system before starting with drawing a use case diagram, and then the system's functionalities are found. And once every single functionality is identified, they are then transformed into the use cases to be used in the use case diagram.

After that, we will enlist the actors that will interact with the system. The actors are the person or a thing that invokes the functionality of a system. It may be a system or a private entity, such that it requires an entity to be pertinent to the functionalities of the system to which it is going to interact.

Once both the actors and use cases are enlisted, the relation between the actor and use case/system is inspected. It identifies the no of times an actor communicates with the system. Basically, an actor can interact multiple times with a use case or system at a particular instance of time.

Following are some rules that must be followed while drawing a use case diagram:

1. A pertinent and meaningful name should be assigned to the actor or a use case of a system.

2. The communication of an actor with a use case must be defined in an understandable way.
3. Specified notations to be used as and when required.

Data Scraping Use Case Diagram



Figure 3.3 Usecase diagram

3.3 Activity Diagram

In UML, the activity diagram is used to demonstrate the flow of control within the system rather than the implementation. It models the concurrent and sequential activities.

The activity diagram helps in envisioning the workflow from one activity to another. It put emphasis on the condition of flow and the order in which it occurs. The flow can be sequential, branched, or concurrent, and to deal with such kinds of flows, the activity diagram has come up with a fork, join, etc.

It is also termed as an object-oriented flowchart. It encompasses activities composed of a set of actions or operations that are applied to model the behavioral diagram.

Why to draw an Activity diagram?

An activity diagram is a flowchart of activities, as it represents the workflow among various activities. They are identical to the flowcharts, but they themselves are not exactly the flowchart. In other words, it can be said that an activity diagram is an enhancement of the flowchart, which encompasses several unique skills. Since it incorporates swimlanes, branching, parallel flows, join nodes, control nodes, and forks, it supports exception handling. A system must be explored as a whole before drawing an activity diagram to provide a clearer view of the user. All of the activities are explored after they are properly analyzed for finding out the constraints applied to the activities. Each and every activity, condition, and association must be recognized.

After gathering all the essential information, an abstract or a prototype is built, which is then transformed into the actual diagram.

Following are the rules that are to be followed for drawing an activity diagram:

1. A meaningful name should be given to each and every activity.
2. Identify all of the constraints.
3. Acknowledge the activity associations.

When to use an Activity Diagram?

An activity diagram can be used to portray business processes and workflows. Also, it is used for modeling business as well as the software. An activity diagram is utilized for the followings:

1. To graphically model the workflow in an easier and understandable way.
2. To model the execution flow among several activities.
3. To model comprehensive information of a function or an algorithm employed within the system.

Data Scraping Activity Diagram -

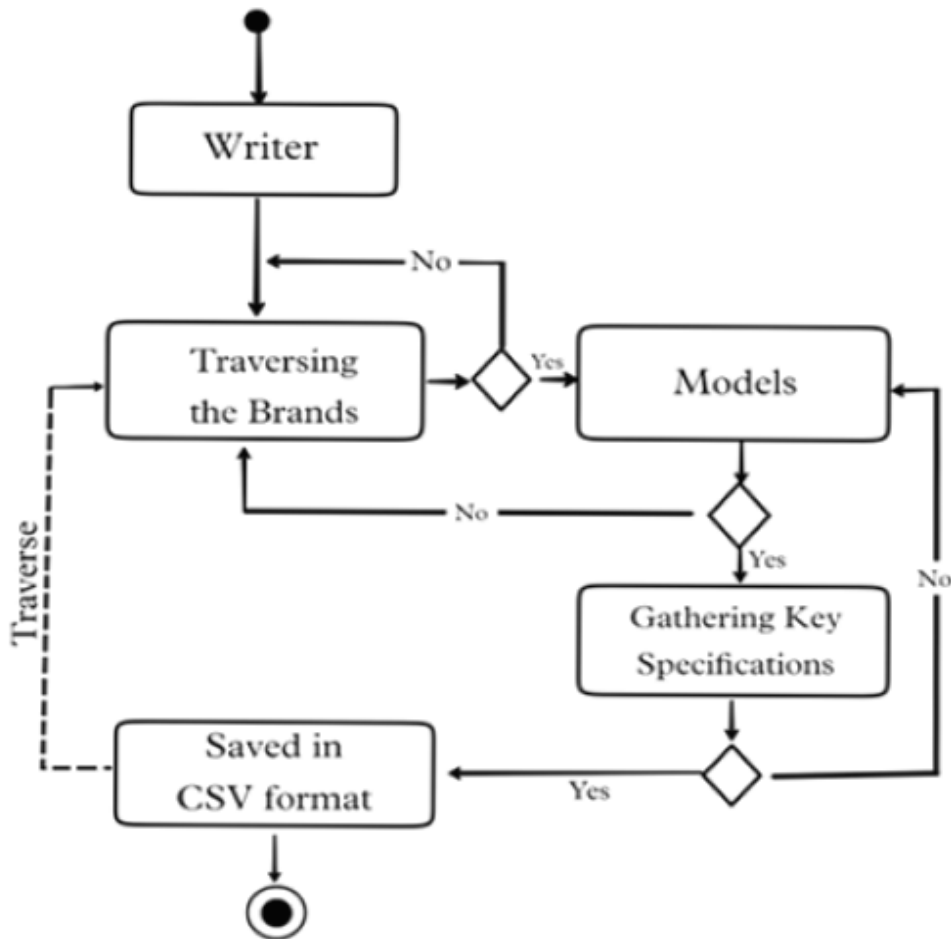


Figure 3.4 Activity diagram

CHAPTER : 4

TECHNOLOGY USED

4.1 Python

Python is a programming language. It was created by Guido van Rossum, and released in 1991.

Python is a computer programming language often used to build websites and software, automate tasks, and conduct data analysis. Python is a general-purpose language, meaning it can be used to create a variety of different programs and isn't specialized for any specific problems. This versatility, along with its beginner-friendliness, has made it one of the most-used programming languages today. A survey conducted by industry analyst firm RedMonk found that it was the second-most popular programming language among developers in 2021.

Python is commonly used for developing websites and software, task automation, data analysis, and data visualization. Since it's relatively easy to learn, Python has been adopted by many non-programmers such as accountants and scientists, for a variety of everyday tasks, like organizing finances.

“Writing programs is a very creative and rewarding activity,” says University of Michigan and Coursera instructor Charles R Severance in his book *Python for Everybody*. “You can write programs for many reasons, ranging from making your living to solving a difficult data analysis problem to having fun to helping someone else solve a problem.”

What can you do with python? Some things include:

- Data analysis and machine learning
- Web development
- Automation or scripting
- Software testing and prototyping

- Everyday tasks

Python is popular for a number of reasons. Here's a deeper look at what makes it so versatile and easy to use for coders.

- It has a simple syntax that mimics natural language, so it's easier to read and understand. This makes it quicker to build projects, and faster to improve on them.
- It's versatile. Python can be used for many different tasks, from web development to machine learning.
- It's beginner friendly, making it popular for entry-level coders.
- It's open source, which means it's free to use and distribute, even for commercial purposes.
- Python's archive of modules and libraries—bundles of code that third-party users have created to expand Python's capabilities—is vast and growing.
- Python has a large and active community that contributes to Python's pool of modules and libraries, and acts as a helpful resource for other programmers. The vast support community means that if coders run into a stumbling block, finding a solution is relatively easy; somebody is bound to have encountered the same problem before.

Import: Python modules can get access to code from another module by importing the file/function using import. The import statement is the most common way to invoking the import machinery, but it is not the only way.

4.1.1 Beautiful Soup.

Beautiful Soup is a Python library that is used for web scraping purposes to pull the data out of HTML and XML files. It creates a parse tree from page source code that can be used to extract data in a hierarchical and more readable manner.

Beautiful Soup supports the HTML parser included in Python's standard library, but it also supports a number of third-party Python parsers. One is the [lxml parser](#). Depending on your setup, you might install lxml with one of these commands:

Beautiful Soup defines a lot of methods for searching the parse tree, but they're all very similar. I'm going to spend a lot of time explaining the two most popular methods: `find()` and `find_all()`. The other methods take almost exactly the same arguments, so I'll just cover them briefly.

Installation

How to install BeautifulSoup

For installing BeautifulSoup we need Python made framework for the same, and also some other supported or additional frameworks can be installed by given PIP command below:

```
pip install beautifulsoup4
```

Beautiful Soup is a great tool for extracting very specific information from large unstructured raw Data, and also it is very fast and handy to use.

Advantage

- Very fast
- Extremely lenient
- Parses pages the same way a Browser does
- Prettify the Source Code

4.1.2 Requests.

The Requests library simplifies making HTTP requests to web servers and working with their responses.

The Requests library provides a simple API for interacting with HTTP operations such as GET, POST, etc.

The methods implemented in the Requests library execute HTTP operations against a specific web server specified by its URL.

It also supports sending extra information to a web server through parameters and headers, encoding the server responses, detecting errors, and handling redirects.

The Hypertext Transfer Protocol (HTTP) is a request/response protocol based on the client-server architecture that relies on TCP/IP connections for exchanging request and response messages.

HTTP clients such as web browsers or mobile applications send requests to an HTTP server, and the server responds to them with messages containing a status line, a header, and a body.

The HTTP request returns a [Response Object](#) with all the response data (content, encoding, status, etc).

To install requests, simply:

```
$ pip install requests
```

Advantages of using Python Requests Library

Following are the advantages of using Python Requests Library –

- Easy to use and fetch the data from the URL given.
- Requests library can be used to scrape the data from the website.

- Using requests, you can get, post, delete, update the data for the URL given.
- The handling of cookies and session is very easy.
- The security is also taken care of the help of authentication module support.

4.1.3 Pandas.

Pandas is defined as an open-source library that provides high-performance data manipulation in Python. The name of Pandas is derived from the word Panel Data, which means an Econometrics from Multidimensional data. It is used for data analysis in Python and developed by Wes McKinney in 2008.

Data analysis requires lots of processing, such as restructuring, cleaning or merging, etc. There are different tools available for fast data processing, such as Numpy, Scipy, Cython, and Panda. But we prefer Pandas because working with Pandas is fast, simple and more expressive than other tools.

Pandas is built on top of the Numpy package, means Numpy is required for operating the Pandas.

Before Pandas, Python was capable for data preparation, but it only provided limited support for data analysis. So, Pandas came into the picture and enhanced the capabilities of data analysis. It can perform five significant steps required for processing and analysis of data irrespective of the origin of the data, i.e., load, manipulate, prepare, model, and analyze.

The Pandas provides two data structures for processing the data, i.e., Series and DataFrame.

Pandas makes it easy to scrape a table(<table>tag) on a web page.

Key Features of Pandas

- It has a fast and efficient DataFrame object with the default and customized indexing.
- Used for reshaping and pivoting of the data sets.
- Group by data for aggregations and transformations.

- It is used for data alignment and integration of the missing data.
- Provide the functionality of Time Series.
- Process a variety of data sets in different formats like matrix data, tabular heterogeneous, time series.

4.2 Excel.

Excel is a spreadsheet program from Microsoft and a component of its Office product group for business applications. Microsoft Excel enables users to format, organize and calculate data in a spreadsheet.

By organizing data using software like Excel, data analysts and other users can make information easier to view as data is added or changed. Excel contains a large number of boxes called cells that are ordered in rows and columns. Data is placed in these cells.

Excel is a part of the Microsoft Office and Office 365 suites and is compatible with other applications in the Office suite. The spreadsheet software is available for Windows, macOS, Android and iOS platforms.

Features of MS Excel

Various editing and formatting can be done on an Excel spreadsheet. Discussed below are the various features of MS Excel.

The image below shows the composition of features in MS Excel:

- **Home**
- Comprises options like font size, font styles, font colour, background colour, alignment, formatting options and styles, insertion and deletion of cells and editing options

-
- **Insert**
- Comprises options like table format and style, inserting images and figures, adding graphs, charts and sparklines, header and footer option, equation and symbols
-
- **Page Layout**
- Themes, orientation and page setup options are available under the page layout option
-
- **Formulas**
- Since tables with a large amount of data can be created in MS excel, under this feature, you can add formulas to your table and get quicker solutions
-
- **Data**
- Adding external data (from the web), filtering options and data tools are available under this category
-
- **Review**
- Proofreading can be done for an excel sheet (like spell check) in the review category and a reader can add comments in this part
-
- **View**
- Different views in which we want the spreadsheet to be displayed can be edited here. Options to zoom in and out and pane arrangement are available under this category

Excel and XLS files

An XLS file is a spreadsheet file that can be created by Excel or other spreadsheet programs. The file type represents an Excel Binary File format. An XLS file stores data as binary streams -- a

compound file. Streams and substreams in the file contain information about the content and structure of an Excel workbook.

Versions of Excel after Excel 2007 use XLSX files by default, since it is a more open and structured format. Later versions of Excel still support the creation and reading of XLS files, however. Workbook data can also be exported in formats including PDF, TXT, Hypertext markup language, XPS and XLSX.

Macro-enabled Excel files use the XLSM file extension. In this case, macros are sets of instructions that automate Excel processes. XLSM files are similar to XLM files but are based on the Open XML format found in later Microsoft Office software.

Benefits of Using MS Excel

MS Excel is widely used for various purposes because the data is easy to save, and information can be added and removed without any discomfort and less hard work.

Given below are a few important benefits of using MS Excel:

- **Easy To Store Data:** Since there is no limit to the amount of information that can be saved in a spreadsheet, MS Excel is widely used to save data or to analyse data. Filtering information in Excel is easy and convenient.
- **Easy To Recover Data:** If the information is written on a piece of paper, finding it may take longer, however, this is not the case with excel spreadsheets. Finding and recovering data is easy.
- **Application of Mathematical Formulas:** Doing calculations has become easier and less time-taking with the formulas option in MS excel

- **More Secure:** These spreadsheets can be password secured in a laptop or personal computer and the probability of losing them is way lesser in comparison to data written in registers or piece of paper.
- **Data at One Place:** Earlier, data was to be kept in different files and registers when the paperwork was done. Now, this has become convenient as more than one worksheet can be added in a single MS Excel file.
- **Neater and Clearer Visibility of Information:** When the data is saved in the form of a table, analysing it becomes easier. Thus, information is a spreadsheet that is more readable and understandable.

4.3 Power BI.

Power BI is a business analytics service provided by Microsoft that lets you visualize your data and share insights. It converts data from different sources to build interactive dashboards and Business Intelligence reports.

Power BI is a Data Visualization and Business Intelligence tool that converts data from different data sources to interactive dashboards and BI reports. Power BI suite provides multiple software, connector, and services - Power BI desktop, Power BI service based on SaaS, and mobile Power BI apps available for different platforms. These set of services are used by business users to consume data and build BI reports.

Power BI desktop app is used to create reports, while Power BI Services (Software as a Service - SaaS) is used to publish the reports, and Power BI mobile app is used to view the reports and dashboards.

Power BI provides a scalable platform that helps the user to connect, visualize, and share the data with other users or stakeholders to gain deeper insights into the business. It is available in both free and paid versions.

Components of Power BI

1. Power Query

Power Query is the data transformation and mash up the engine. It enables you to discover, connect, combine, and refine data sources to meet your analysis need. It can be downloaded as an add-in for Excel or can be used as part of the Power BI Desktop.

2. Power Pivot

Power Pivot is a data modeling technique that lets you create data models, establish relationships, and create calculations. It uses Data Analysis Expression (DAX) language to model simple and complex data.

3. Power View

Power View is a technology that is available in Excel, Sharepoint, SQL Server, and Power BI. It lets you create interactive charts, graphs, maps, and other visuals that bring your data to life. It can connect to data sources and filter data for each data visualization element or the entire report.

4. Power Map

Microsoft's Power Map for Excel and Power BI is a 3-D data visualization tool that lets you map your data and plot more than a million rows of data visually on Bing maps in 3-D format from an Excel table or Data Model in Excel. Power Map works with Bing maps to get the best visualization based on latitude, longitude, or country, state, city, and street address information.

5. Power BI Desktop

Power BI Desktop is a development tool for Power Query, Power Pivot, and Power View. With Power BI Desktop, you have everything under the same solution, and it is easier to develop BI and data analysis experience.

6. Power Q&A

The Q&A feature in Power BI lets you explore your data in your own words. It is the fastest way to get an answer from your data using natural language. An example could be what was the total sales last year? Once you've built your data model and deployed that into the Power BI website, then you can ask questions and get answers quickly.

Advantages of Power BI

Affordability:

A major advantage of using Power BI is that it is inexpensive compared to other cloud service providers. The Power BI Desktop version is free of cost, you can download and start making the reports on your computer. However, if you wanna share your reports on the cloud you have to pay 9.99\$ per user per month.

Excel Integration:

In Power BI, you can also save data to Excel. No matter how great the data is presented using Graphs, maps and charts using data visualization tools, people still tend to have the data in their excel sheet. For example, you can get the data of a manufacturing unit for the past six months within a few clicks from Power BI.

Custom Visualization:

Power BI offers a wide range of custom visualizations where developers can take your requirements and convert them to KPI's, charts, graphs, maps etc.

Data Accessibility and Interactive visualization:

Power BI provides great access to all the data source and the data sets that you create while designing the reports, in a centralized location. You can access the data anytime, anywhere from

any device multiple times. Users can interact with the dashboards using filters, highlighting features, etc. by simple clicks.

Newly Developed Features:

One of the other advantages is that Microsoft provides the users with monthly updates. Overall, Power BI is an amazing tool for data analysis and visualization. The pros far outbalance the drawbacks of Power BI.

Slicer: Slicer is a way of filtering. They narrow the portion of the dataset that is shown in the order report visualization.


Slicers are a great choice when you want to:

Display commonly used or important filters on the report canvas for easier access.

Make it easier to see the current filtered state without having to open a drop-down list.

Filter by columns that are unneeded and hidden in the data tables.

Create more focused reports by putting slicers next to important visuals.



Bentley	Kia	MINI	Skoda
Citroen	Lamborghini	Nissan	Tata
Jaguar	Maserati	Porsche	Volkswagen
Jeep	McLaren	Renault	Volvo

Figure 4.1 Slicer

Power BI slicers don't support:

Input fields

Drill-down options

Pi Chart

The purpose of a Pie chart is to illustrate the contribution of different values to a total.

For example, to see the total sales split by product category. You can then see the percentage contribution of each product category to the total revenue.

The Pie chart is not the only chart type that can produce this visual. We could opt for a Donut chart, or maybe a Column chart instead.

It is important to be able to create different chart types, as you may be asked by someone to display data in a specific way.

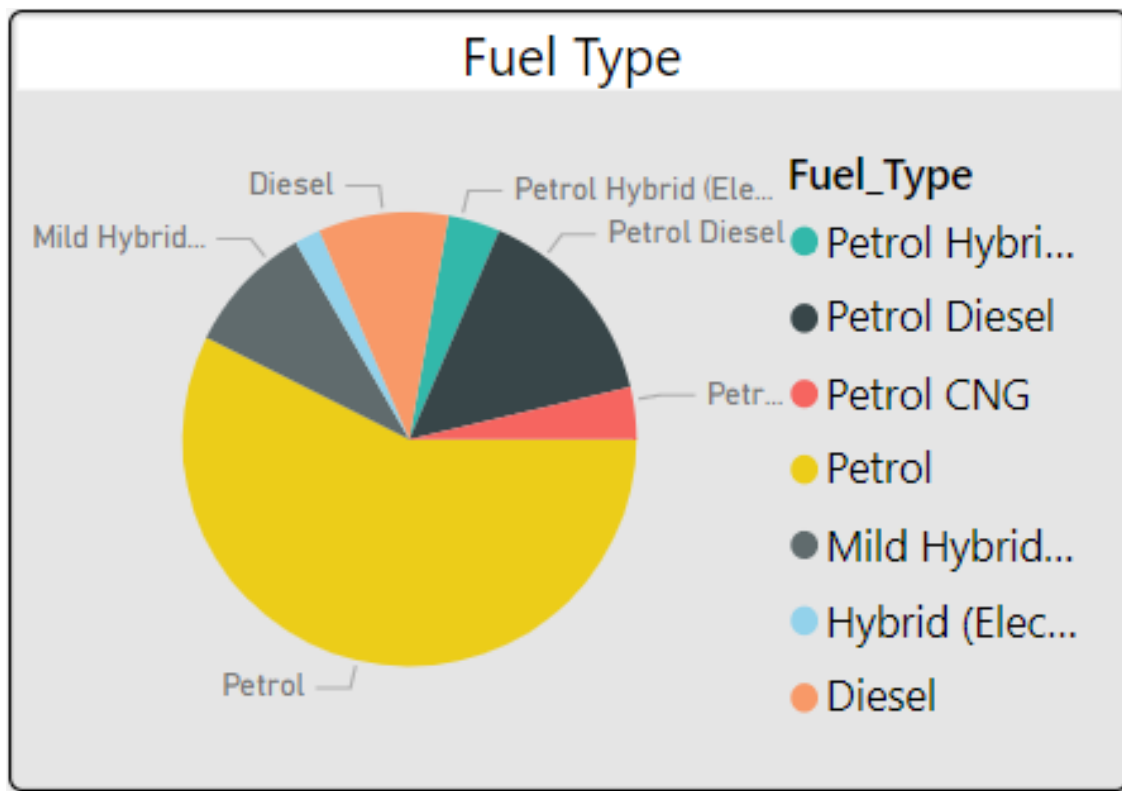


Figure 4.2 Pi chart

Bar and Column Charts:

Bar and column charts are some of the most widely used visualization charts in Power BI. They can be used for one or multiple categories. Both these chart types represent data with rectangular bars, where the size of the bar is proportional to the magnitude of data values

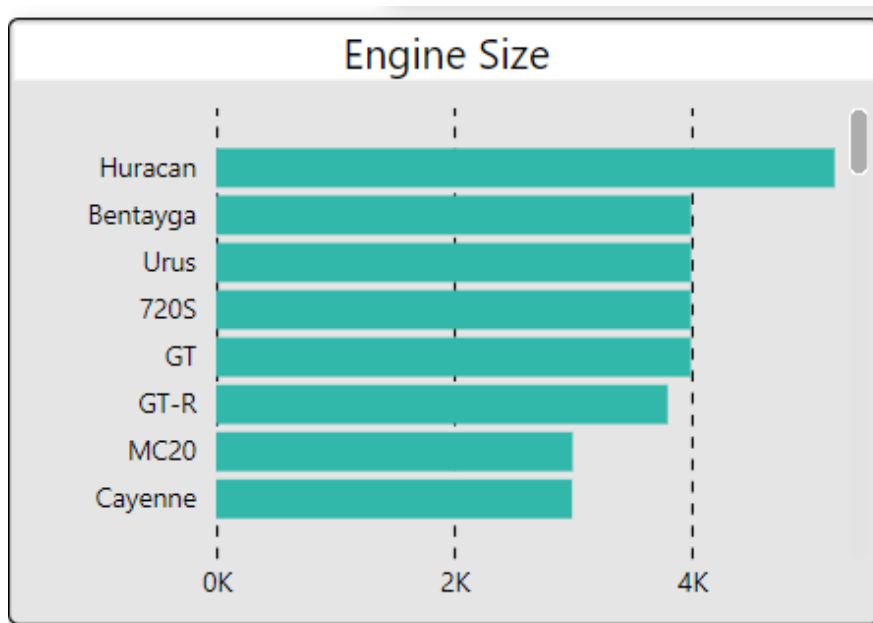


Figure 4.3 Bar chart

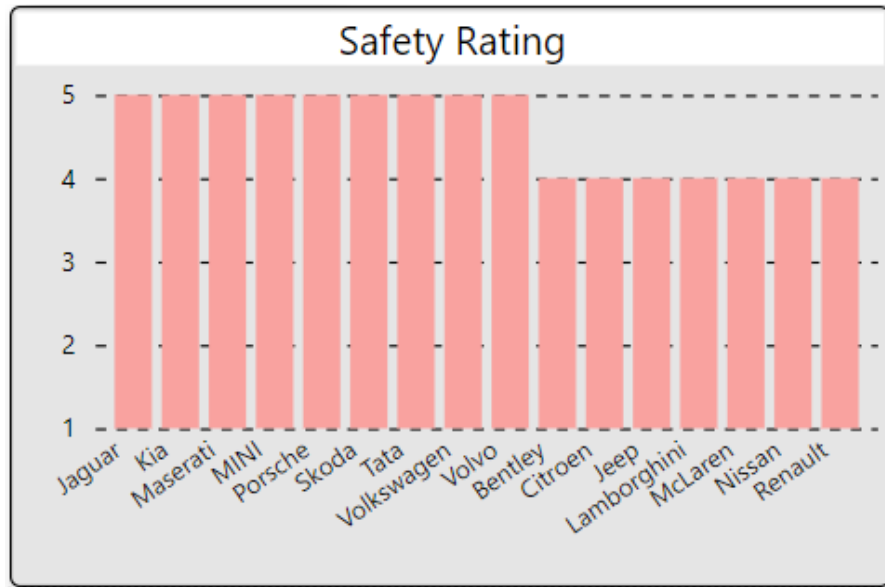


Figure 4.4 Column chart

Card

A card displays the aggregated information of a single numeric measure value. However, you can change the aggregate function from default sum to Avg, min, etc. A card is the best way to represent the overall numerical information of any measure. For instance, total sales, profit, orders, etc.

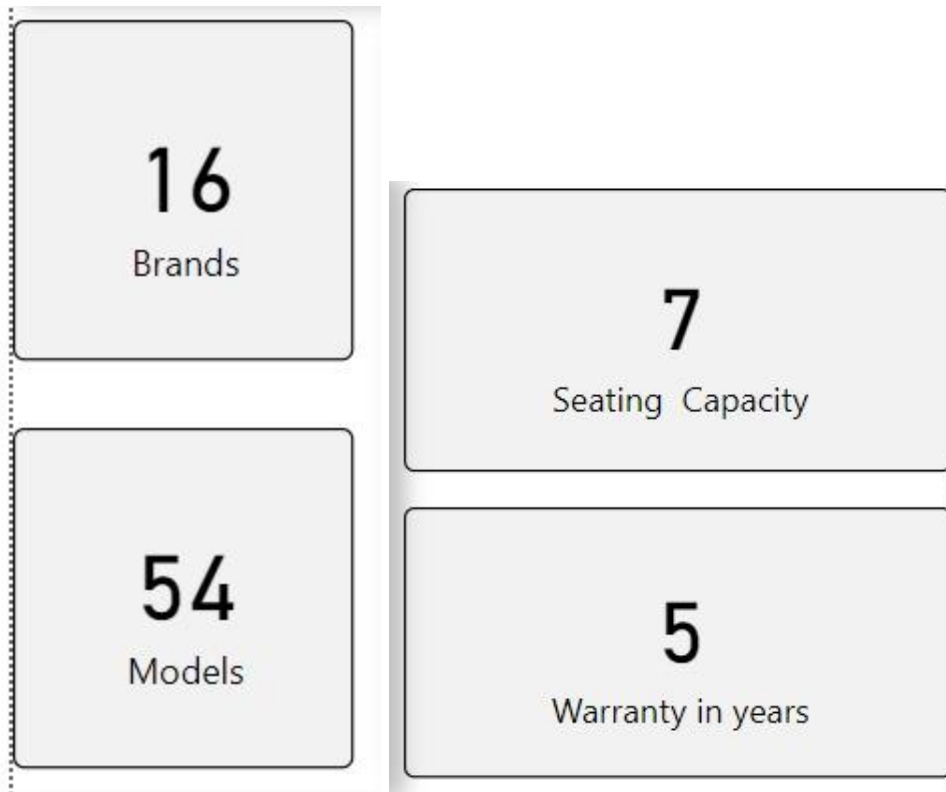


Figure 4.5 Card

CHAPTER- 5

CODING AND IMPLEMENTATION

Code-

```
# importing required libraries
import requests
from bs4 import BeautifulSoup
from csv import writer
import pandas

# Selected Brands Name
a =
["Tata","Kia","Volkswagen","Skoda","Jaguar","Renault","Jeep","Volvo","Nissan","Lamborghini",
,"Porsche","MINI","Citroen","Maserati","Bentley","McLaren"]

with open("CarsFinal.csv","w",encoding="utf8", newline="")as file:
    Writer = writer(file)
    header =
["Brand_Name","Model_Name","Price","Fuel_Type","Mileage","Seating_Capacity","Safety_Ra
ting","Warranty","Engine_Size","Transmission","Size","Fuel_Tank"]
    Writer.writerow(header)

for Brand_name in a:
    # Store website in a variable
    url2 = "https://www.cartrade.com/" + Brand_name + "-cars/"
    print(fThis is the website: {url2}')

    # HTTP request
    r = requests.get(url2)
    r.status_code

    soup = BeautifulSoup(r.content,'html.parser')

    link1 = "carBrands_list"
    cars = soup.find_all('div', class_=link1)
    # cars
```

```
len(cars)
```

```
c = []
for name in cars:
    a = name.find("div", class_="makeblklitl")

    if a != None:
        b = a.get_text()
        # print(b)
        if b != None:
            c.append(b)
print(c)
print("This are the names of all models")
```

```
x = []
for dif in c:
    dif = dif.split()
    x.append(dif)
print(x, " here")
```

```
for model_name in x:
    url3 = ("https://www.cartrade.com/"+Brand_name+"-cars/"+model_name[1]+"/")
    # print(url3)
    # print(model_name)

    r = requests.get(url3)
    r.status_code

    soup = BeautifulSoup(r.content, 'html.parser')

    link3 = "keySpecsBody"
    cars = soup.find('tbody', class_=link3).find_all('tr')
    # cars = soup.find('tbody', class_=link3)
    # print(cars)

    if cars != None :
```

```

Brand_Name    = Brand_name
Model_Name    = model_name[1]
Price         = cars[0].find("td").get_text()
Fuel_Type     = cars[1].find("td").get_text()
Mileage       = cars[2].find("td").get_text()
Seating_Capacity = cars[3].find("td").get_text()
Safety_Rating = cars[4].find("td").get_text()
Warranty      = cars[5].find("td").get_text()
Engine_Size   = cars[6].find("td").get_text()
Transmission  = cars[7].find("td").get_text()
Size          = cars[8].find("td").get_text()
Fuel_Tank     = cars[9].find("td").get_text()

Info
=[Brand_Name,Model_Name,Price,Fuel_Type,Mileage,Seating_Capacity,Safety_Rating,Warrant
y,Engine_Size,Transmission,Size,Fuel_Tank]

print('done')
print(Info)

Writer.writerow(Info)
print("Done")

```

Excel Sheets

	A	B	C	D	E	F	G	H	I	J	K
1	Brand_Name	Model_Name	Price	Fuel_Type	Mileage	Seating_Capacity	Safety_Rating	Warranty	Engine_Size in cc	Transmission	Size
2	Tata	Tiago	5.44 - 7.90 Lakh	Petrol CNG	20 - 26.4 km/l	5	4	2	1199	Manual Automatic (AMT)	3765 mm L X 1677 mm W X 1535 mm H
3	Tata	Punch	6.00 - 9.54 Lakh	Petrol	18.8 - 18.9 km/l	5	5	2	1199	Manual Automatic (AMT)	3827 mm L X 1742 mm W X 1615 mm H
4	Tata	Tigor	6.09 - 8.84 Lakh	Petrol CNG	19.2 - 26.4 km/l	5	4	2	1199	Manual Automatic (AMT)	3993 mm L X 1677 mm W X 1532 mm H
5	Tata	Altroz	6.34 - 10.25 Lakh	Petrol Diesel	18.1 - 23 km/l	5	5	2	1199	Manual Automatic (DCT)	3990 mm L X 1755 mm W X 1523 mm H
6	Tata	Tiago	5.44 - 7.90 Lakh	Petrol CNG	20 - 26.4 km/l	5	4	2	1199	Manual Automatic (AMT)	3765 mm L X 1677 mm W X 1535 mm H
7	Tata	Nexon	7.69 - 14.18 Lakh	Petrol Diesel	16.3 - 22 km/l	5	5	2	1199	Manual Automatic (AMT)	3993 mm L X 1811 mm W X 1606 mm H
8	Tata	Harrier	14.79 - 22.34 Lakh	Diesel	14.6 - 16.3 km/l	5	4	2	1956	Manual Automatic (TC)	4598 mm L X 1894 mm W X 1706 mm H
9	Tata	Safari	15.45 - 23.76 Lakh	Diesel	14 - 16.1 km/l	4	4	2	1956	Manual Automatic (TC)	4661 mm L X 1894 mm W X 1786 mm H
10	Kia	Sonet	7.49 - 13.99 Lakh	Petrol Diesel	18.2 - 24.1 km/l	5	4	3	1197	Manual Clutchless Manual (IMT) Aut	3995 mm L X 1790 mm W X 1610 mm H
11	Kia	Carens	10.00 - 18.00 Lakh	Petrol Diesel	15.7 - 21.3 km/l	4	4	3	1353	Manual Automatic (DCT) Automatic	4540 mm L X 1800 mm W X 1708 mm H
12	Kia	Seltos	10.49 - 18.65 Lakh	Petrol Diesel	16.1 - 21 km/l	5	3	3	1493	Manual Clutchless Manual (IMT) Aut	4315 mm L X 1800 mm W X 1620 mm H
13	Kia	Carnival	29.99 - 34.98 Lakh	Diesel	13.9 km/l	4	5	3	2199	Automatic (TC)	5115 mm L X 1985 mm W X 1755 mm H
14	Volkswagen	Virtus	11.32 - 18.42 Lakh	Petrol	18.1 - 19.4 km/l	5	4	4	999	Manual Automatic (TC) Automatic (E 4561 mm L X 1752 mm W X 1507 mm H	
15	Volkswagen	Taigun	11.56 - 18.71 Lakh	Petrol	17.2 - 19.2 km/l	5	5	4	999	Manual Automatic (TC) Automatic (E 4221 mm L X 1760 mm W X 1612 mm H	
16	Volkswagen	Tiguan	33.50 Lakh	Petrol	12.6 km/l	5	5	4	1984	Automatic (DCT)	4509 mm L X 1839 mm W X 1665 mm H
17	Skoda	Slavia	11.29 - 18.40 Lakh	Petrol	18 - 19.4 km/l	5	4	4	999	Manual Automatic (TC) Automatic (E 4541 mm L X 1752 mm W X 1507 mm H	
18	Skoda	Kushaq	11.58 - 19.69 Lakh	Petrol	17.2 - 19.2 km/l	5	5	4	999	Manual Automatic (TC) Automatic (E 4225 mm L X 1760 mm W X 1612 mm H	
19	Skoda	Octavia	27.34 - 30.44 Lakh	Petrol	15.8 km/l	5	5	4	1984	Automatic (DCT)	4689 mm L X 1829 mm W X 1469 mm H
20	Skoda	Superb	34.16 - 37.26 Lakh	Petrol	15.1 km/l	5	5	4	1984	Automatic (DCT)	4869 mm L X 1864 mm W X 1469 mm H
21	Skoda	Kodiahq	37.49 - 39.99 Lakh	Petrol	12.7 km/l	7	5	4	1984	Automatic (DCT)	4699 mm L X 1882 mm W X 1665 mm H
22	Jaguar	XF	71.60 - 76.00 Lakh	Petrol Diesel	14.4 - 19.2 km/l	5	5	3	1997	Automatic (TC)	4962 mm L X 1982 mm W X 1456 mm H
23	Jaguar	F-Pace	74.88 Lakh	Petrol Diesel	12.9 - 19.3 km/l	5	5	3	1997	Automatic (TC)	4747 mm L X 2071 mm W X 1664 mm H
24	Jaguar	F-Type	97.93 Lakh - 2.61 Crore	Petrol	9.1 - 12.3 km/l	2	4	3	1997	Automatic (TC)	4470 mm L X 1923 mm W X 1311 mm H
25	Renault	Kwid	4.64 - 6.09 Lakh	Petrol	20.7 - 22 km/l	5	1	2	799	Manual Automatic (AMT)	3731 mm L X 1579 mm W X 1474 mm H
26	Renault	Triber	5.90 - 8.51 Lakh	Petrol	18.2 - 19 km/l	7	4	2	999	Manual Automatic (AMT)	3991 mm L X 1739 mm W X 1643 mm H
27	Renault	Kiger	5.99 - 10.62 Lakh	Petrol	18.2 - 20.5 km/l	5	4	2	999	Manual Automatic (AMT) Automatic	3991 mm L X 1750 mm W X 1605 mm H
28	Jeep	Compass	19.27 - 32.67 Lakh	Petrol Diesel	13.8 - 17.3 km/l	5	4	3	1368	Manual Automatic (DCT) Automatic	4405 mm L X 1818 mm W X 1640 mm H
29	Jeep	Meridian	29.90 - 36.95 Lakh	Diesel	14.9 - 16.2 km/l	7	4	3	1956	Manual Automatic (TC)	4769 mm L X 1859 mm W X 1698 mm H

Figure 5.1 Excel sheet

	A	B	C	D	E	F	G	H	I	J	K
29	Jeep	Meridian	29.90 - 36.95 Lakh	Diesel	14.9 - 16.2 km/l	7	4	3	1956	Manual Automatic (TC)	4769 mm L X 1859 mm W X 1698 mm H
30	Jeep	Wrangler	57.83 - 61.83 Lakh	Petrol	12.1	5	4	2	1998	Automatic (TC)	4882 mm L X 1894 mm W X 1838 mm H
31	Volvo	XC40	45.85 Lakh	Mild Hybrid(Electr	14.4 km/l	5	5	2	1969	Automatic	4440 mm L X 1863 mm W X 1652 mm H
32	Volvo	S60	45.90 Lakh	Mild Hybrid(Electr	14 km/l	5	5	2	1969	Automatic (TC)	4761 mm L X 2040 mm W X 1431 mm H
33	Volvo	XC60	65.90 Lakh	Mild Hybrid(Electr	12.4 km/l	5	5	2	1969	Automatic	4708 mm L X 1902 mm W X 1653 mm H
34	Volvo	S90	66.90 Lakh	Mild Hybrid(Electr	14.7 km/l	5	5	2	1969	Automatic	4969 mm L X 1879 mm W X 1440 mm H
35	Volvo	XC90	94.90 Lakh	Mild Hybrid(Electr	11.04 km/l	7	5	2	1969	Automatic	4953 mm L X 2008 mm W X 1773 mm H
36	Nissan	Magnite	5.97 - 10.53 Lakh	Petrol	17.7 - 20 km/l	5	4	2	999	Manual Automatic (CVT)	3994 mm L X 1758 mm W X 1572 mm H
37	Nissan	Kicks	9.50 - 14.88 Lakh	Petrol	13.9 - 15.8 km/l	5	3	2	1498	Manual Automatic (CVT)	4384 mm L X 1813 mm W X 1669 mm H
38	Nissan	GT-R	2.12 Crore	Petrol	8.4 km/l	4	4	3	3799	Automatic (DCT)	4710 mm L X 1895 mm W X 1370 mm H
39	Lamborghini	Urus	3.10 Crore	Petrol	7.8 km/l	5	4	3	3996	Automatic (DCT)	5112 mm L X 2016 mm W X 1638 mm H
40	Lamborghini	Huracan	3.22 - 3.73 Crore	Petrol	7.2 - 7.3 km/l	2	4	3	5204	Automatic (DCT)	4520 mm L X 2236 mm W X 1165 mm H
41	Porsche	Macan	83.21 Lakh	Petrol	11.4 km/l	5	5	3	1984	Automatic (DCT)	4725 mm L X 1922 mm W X 1621 mm H
42	Porsche	Cayenne	1.26 - 1.93 Crore	Petrol Hybrid (Elec	8.6 - 41.6 km/l	5	5	5	2995	Automatic (TC)	4918 mm L X 1983 mm W X 1696 mm H
43	Porsche	718	1.32 - 2.54 Crore	Petrol	7.5 - 14.5 km/l	2	4	3	1988	Automatic (DCT) Manual	4379 mm L X 1801 mm W X 1295 mm H
44	Porsche	Panamera	1.57 - 2.73 Crore	Petrol Hybrid (Elec	9.7 - 30.3 km/l	4	4	2	2894	Automatic (DCT)	5049 mm L X 1937 mm W X 1423 mm H
45	Porsche	911	1.72 - 3.25 Crore	Petrol	7.4 - 11.2 km/l	4	4	3	2981	Automatic (DCT) Manual	4519 mm L X 1852 mm W X 1298 mm H
46	MINI	Cooper	40.00 - 40.58 Lakh	Petrol	16.3 km/l	4	4	2	1998	Automatic (DCT)	3850 mm L X 1727 mm W X 1414 mm H
47	MINI	COUNTRYMAN	46.00 Lakh	Petrol	14.3 km/l	5	5	2	1998	Automatic (DCT)	4299 mm L X 1822 mm W X 1557 mm H
48	Citroen	C3	5.88 - 8.15 Lakh	Petrol	19.4 - 19.8 km/l	5	4	2	1198	Manual	3981 mm L X 1733 mm W X 1586 mm H
49	Citroen	C5	32.74 - 33.78 Lakh	Diesel	18.6 km/l	5	4	3	1997	Automatic (TC)	4500 mm L X 1969 mm W X 1710 mm H
50	Maserati	Ghibli	1.20 - 1.99 Crore	Hybrid (Electric + I	8 - 11.4 km/l	5	5	4	1998	Automatic (TC)	4971 mm L X 1945 mm W X 1461 mm H
51	Maserati	Levante	1.45 - 2.38 Crore	Petrol	9.3 - 9.8 km/l	5	4	3	2979	Automatic (TC)	5005 mm L X 1981 mm W X 1693 mm H
52	Maserati	Quattroporte	1.80 - 2.32 Crore	Petrol	8.2 - 9.4 km/l	5	5	3	2979	Automatic (TC)	5262 mm L X 1948 mm W X 1481 mm H
53	Maserati	MC20	3.65 Crore	Petrol	8.6 km/l	2	4	4	3000	Automatic (DCT)	4669 mm L X 2178 mm W X 1224 mm H
54	Bentley	Bentayga	4.10 Crore	Petrol	7.6 km/l	5	4	3	3996	Automatic (TC)	5125 mm L X 2222 mm W X 1728 mm H
55	McLaren	GT	3.72 Crore	Petrol	7 km/l	2	4	3	3994	Automatic (DCT)	4683 mm L X 2045 mm W X 1286 mm H
56	McLaren	720S	4.65 - 5.04 Crore	Petrol	8.2 km/l	2	4	3	3994	Automatic (DCT)	4543 mm L X 2059 mm W X 1196 mm H

Figure 5.2 Excel sheet

Power BI Dashboard

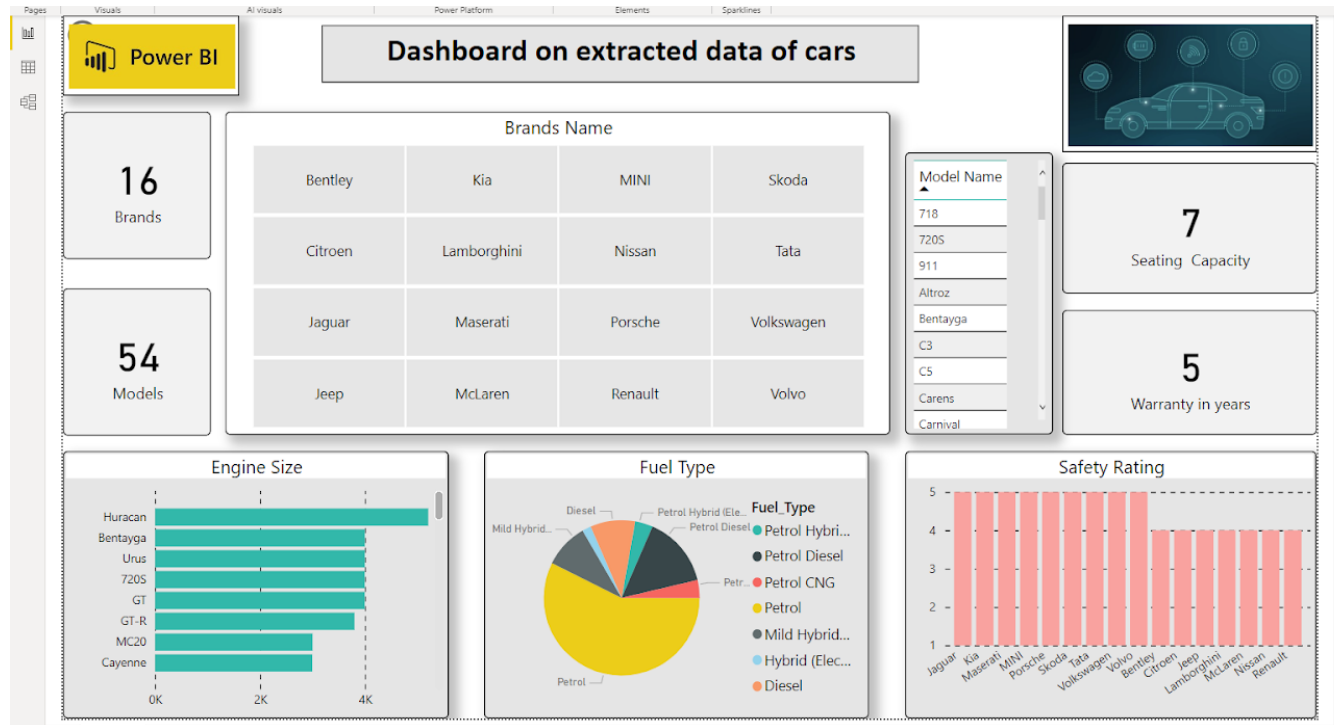


Figure 5.3 Power BI Dashboard

CHAPTER 6

FUTURE ENHANCEMENT

A lot of future work can be done in this area. Developers are just now discovering the potential behind web scrapers and the one developed for this project is very simple.

Future work can be done on building an automated web scraper where you user can press what they want to scrape from the website. A user interface can also be developed for the web scraper so that the user can easily understand how the web scraper work.

In the near future, Web scraping will be one of the important tools in the lead generation process. The web scraping tool can make market research of the particular product/services and enormous benefits to offer in the marketing field.

- Increasing the number of data-sets to increase the amount and accuracy of data in the results.
- Can illustrate via graphics in future, by creating the possibilities of extracting graphics.
- Can work on the overall optimization of of the project by reducing the time.
- Can work on using more websites for a variety and the quantity of the data to be extracted.

CHAPTER : 7

CONCLUSION

The purpose of this minor project was to extract data and visualize it in form of graphs and charts in Power BI application. To systematically arrange the data are provided for research purposes. This program requires good knowledge in python libraries like Request, Beautifulsoup, Pandas and application like Excel and Power BI.

Web scraping allows us to get access to any website on the internet. Implementing web scraping into your business practices could give your business an edge over your competitors. While it gives us unlimited access to any website, necessary care should be taken to avoid misuse of the data. A proper planned and executed web scraping project can help the companies to provide meaningful and rich data to the end user.

Data has become the basis of all decision-making processes whether it's a business or a non-profit organization. Therefore, web scraping has found its applications in every endeavour of note in contemporary times.

CHAPTER- 8

BIBLIOGRAPHY

www.geeksforgeeks.org

www.cartrade.com

www.powerbi.microsoft.com