

Wheels Probe

Submitted By:

Bisma Fajar

Submitted To:

Miss Mahrukh Khan

Fellowship Name:

Bytewise Data Science

Task 7:

Mini Python Project

Contents

1	GitHub Link	4
2	Over View Of Project	4
3	Attributes of Web Scraped Data	5
4	Websites Scraped	6
4.1	Pak Wheels	6
4.2	Garilo	6
4.3	OLX	6
4.4	Gari	7
5	Sorting Algorithm Details	7
5.1	Comparison Based Algorithms	7
5.1.1	Bubble Sort	7
5.1.2	Quick Sort	7
6	Searching Algorithm	8
6.1	Linear Search	8
7	Working Flow Diagram	8
8	FrameWork	8
8.1	Classes	8
8.2	User Interfaces	9
8.3	.py Files	9
9	User Interface	9
9.1	Home Page	9
9.2	Sorting Page	10
9.3	Information Page	11
9.4	Searching Page	11
10	Conclusion	11

List of Figures

1	Data Scraping	6
2	Flow Diagram	8
3	Home Page	9
4	Sorting Window	10
5	Sorting Window	10
6	Information Window	11
7	Searching Window	11

Wheels Probe

1 GitHub Link

<https://github.com/bismafajar816/Data-Science-\BWF-Bisma-Fajar/tree/main/Task7/Wheels%20Probe>

2 Over View Of Project

This task project is a comprehensive data management system that encompasses web scraping from different websites, applying sorting and searching algorithms and making user friendly interface.

To begin with I utilize web scraping techniques to gather and uphold the precision and dependability of this dataset. This approach automates the retrieval of written content, images and other pertinent information guaranteeing current data repository.

To enhance user engagement and facilitate data exploration I have created a user interface that's intuitive and visually attractive. This interface ensures a user experience by allowing users to enter search queries utilize sorting options and access the data in an organized format. This goal is to deliver an experience, for users with levels of technical skills.

Sorting algorithms have an impact, on organizing and analyzing data. I use sorting algorithms like QuickSort and bubble sort to allow users to sort data in either ascending or descending order depending on the attributes

they choose. This functionality gives users the ability to effortlessly identify trends and patterns, within the dataset.

Furthermore for those who prefer a search experience there is an option to perform a single level search. This allows users to conduct keyword searches or filter data based on an attribute.

In conclusion ,the aim of This project is to streamline data retrieval and empower the users with the means to perform analysis and exploration in depth.

3 Attributes of Web Scraped Data

- **Name**
Name of car
- **Model**
Making Year
- **City**
Location where the car is present
- **Price**
Price demanded by seller
- **Kilometers**
No. of km the car has travelled
- **CC**
Engine Specification
- **Fuel**
Type of fuel on which the car performs function like

petrol , diesel or electric

- **Mode**

Specific mode like automatic or manual

4 Websites Scraped

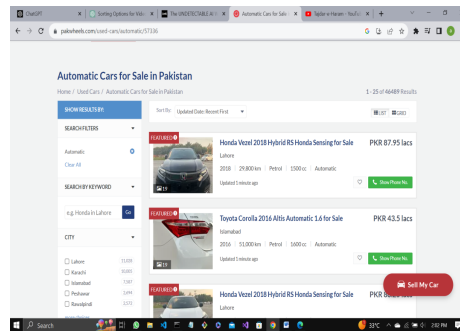


Figure 1: Data Scraping

4.1 Pak Wheels

”Pak Wheels” is a popular online platform in Pakistan which provides comprehensive information about cars ,motorbikes and other vehicles to customers.I scraped about 5 lac data from this website.

4.2 Garilo

”Garilo” is also an online platform in Pakistan.This platform provided limited amount of data to their customer. From this website I scraped only 300 data set.

4.3 OLX

”OLX” is a famous online platform which not only have the information about the automobiles but also provide

information on different clothing and beauty products etc. About 15K data was scraped from this website.

4.4 Gari

”Gari” is online platform of Pakistan which provide lalth of information on automobiles. About 3 lac data was scraped from this website.

5 Sorting Algorithm Details

5.1 Comparison Based Algorithms

5.1.1 Bubble Sort

It sorts a list of elements based on a specified column (attribute) in ascending order. If the specified column is numeric (column 3 or 4), it performs the comparison as a floating-point comparison. This was done because it was throwing errors and not sorting the floats and integers correctly. The algorithm iterates through the data, comparing adjacent elements and swapping them if they are out of order. If no swaps occur in a pass, the data is considered sorted, and the algorithm terminates.

5.1.2 Quick Sort

It can sort in both ascending and descending order, determined by the ”order” parameter. The partition function reorganizes elements around a pivot element, while the quicksort function recursively sorts the data within a given range.

6 Searching Algorithm

6.1 Linear Search

It works by traversing the data set and finding the word entered by the user according to the three provided filters. The user can find the data set starting with required word, ending with the required word or containing that word.

7 Working Flow Diagram

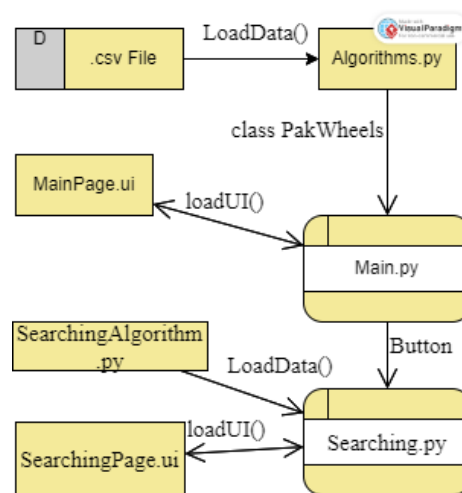


Figure 2: Flow Diagram

8 FrameWork

8.1 Classes

I've written all the algorithms in python files by making classes. Then I've imported the classes and used them in This project.

8.2 User Interfaces

As there was no requirement of user interfaces I simply designed ui using CLI. At first I created user interfaces with Kivy library then it was not suitable for my project and I moved to CLI again.

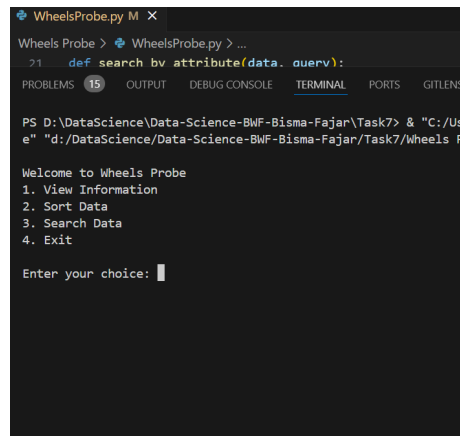
8.3 .py Files

These files are the main files of This project in which I linked backend and frontend of This project. The methods like connecting combo boxes to perform required actions by pushing the push buttons are also written here.

9 User Interface

This project consists of four ui pages, first is Home page, second is information page, third is Sorting page and last is Searching page.

9.1 Home Page



```
WheelsProbe.py M X
Wheels Probe > WheelsProbe.py > ...
71 def search by attribute(data, query):
PROBLEMS 15 OUTPUT DEBUG CONSOLE TERMINAL PORTS GITLENS

PS D:\DataScience\Data-Science-BWF-Bisma-Fajar\Task7> & "C:/Use
e" "d:/DataScience/Data-Science-BWF-Bisma-Fajar/Task7/Wheels Pr

Welcome to Wheels Probe
1. View Information
2. Sort Data
3. Search Data
4. Exit

Enter your choice: |
```

Figure 3: Home Page

9.2 Sorting Page

```
Select Sorting Algorithm
1. Bubble Sort
2. Quick Sort
Enter choice: 1

Select Attribute to Sort By
1. Name
2. Price
3. City
4. Model
5. Kilometers
6. Fuel
7. CC
8. Mode
Enter choice: 1

Data sorted by Name using Bubble Sort
Name: DFSK Glory 580 2021 Pro
Price: 54.75
City: Lahore
Model: 2021
Kilometers: 25,800 km
Fuel: Petrol
CC: 1498 cc
Mode: Automatic
```

Figure 4: Sorting Window

```
Data sorted by Name using Bubble Sort
Name: DFSK Glory 580 2021 Pro
Price: 54.75
City: Lahore
Model: 2021
Kilometers: 25,800 km
Fuel: Petrol
CC: 1498 cc
Mode: Automatic

Name: Daihatsu Hijet 2014 Deluxe
Price: 17.75
City: Karachi
Model: 2014
Kilometers: 140,000 km
Fuel: Petrol
CC: 660 cc
Mode: Automatic

Name: Honda City 2005 i-DSI Vario
Price: 14
City: Lahore
Model: 2005
Kilometers: 275,000 km
Fuel: Petrol
CC: 1300 cc
Mode: Automatic
```

Figure 5: Sorting Window

9.3 Information Page

```
Welcome to Wheels Probe
1. View Information
2. Sort Data
3. Search Data
4. Exit

Enter your choice: 1

Information about Wheels Probe
This application helps you to sort and search car data.
```

Figure 6: Information Window

9.4 Searching Page

```
Enter your choice: 3

Enter search query: Toyota

Search results for "Toyota":
Name: Toyota Corolla 2015 GLi Automatic 1.3 VVTi
Price: 36.9
City: Lahore
Model: 2015
Kilometers: 136,000 km
Fuel: Petrol
CC: 1300 cc
Mode: Automatic

Name: Toyota Corolla 2020 GLi Automatic 1.3 VVTi
Price: 47
City: Jhelum
Model: 2020
Kilometers: 39,000 km
Fuel: Petrol
CC: 1300 cc
Mode: Automatic

Name: Toyota Corolla 2021 Altis Grande X CVT-i 1.8 Black Interior
Price: 67.5
City: Karachi
Model: 2021
Kilometers: 18,939 km
```

Figure 7: Searching Window

10 Conclusion

In conclusion this project helped me to practice my python techniques to do web scraping and enhance my knowledge. Gave a hands on experience how to efficiently apply sorting algorithms for sorting a large data set. I also learned how to use Kivy designer to make graphical interface. And last but not the least it also helped us how to use overleaf.com to make reports and other writings.