**Second Hand Cars Analysis Project**

This project helps most people in real life to save money buying a used Car as it saves almost 50% lower than new cars, no  exaggerated fees, lower insurance, and lower annual registration fees.

it contains most relevant information that provides on car sales including columns: Car name, year, Sell Price, Driven - km, Owner , Mileage, and etc.

* **Start with a question:**

Do people need help to get analyzed data about Used Cars before Buying? In my point **YES**, most people need help to get analyzed data about the used cars before buying them so It’s important to know the price and kms driven on the car as well as the brand to know if the car is a good purchase for second hand owner or not. This data can help many people reach to that conclusion.

 I intend to develop an accurate and efficient system, using modeling and it’s tools , based on a multi-task approach for the second hand cars market

**Tools**

* **Dataset**

its Regression data and its contain 13 columns and 8127 rows (car-details?select=Car\_details.csv) from <https://www.kaggle>[.com/datasets](https://www.kaggle.com/datasets)

* **Libraries**

**Data processing:** pandas ,numpy

**Visualization:** matplotlib ,seaborn , plotly

**Modeling**: sklearn

**programing language**  :Python

* **Modling**

**Linear Regression**

**Steps**

**1.import Libraries**

**2.read\_data**

**3.Explore data analysis**

We explore the data by see data type of each column

and check if there is null values (found **the last 5 columns has null values)**

And the summary statistics using describe ….

**3.Cleaning**

* Clean dataset using some method like drobing the null valuse or fill it
* Remove the duplicated (we found 1202)

after we clean our dataset the num of the columns and rows changed (6925, 11)

**4.visualization**

**bar chart**

**histogram**

**count plot**

**heatmap**

**pair plot**

**5.Modeling**

* **preprocessing for modeling**

convert character columns to dummy variables to be ready for the modeling df=pd.get\_dummies(df)

* **split Data**

detect input and output and split train test

* **feature scaling**

scale all the feature to same scale

* **apply LinearRegression model**

**6.Model evaluation**

# R-Squared

linreg\_model.score(x, y) 0.97