How the development of this activity has helped me to improve my skills into he domain of data science?

This activity was a transformative immersion into the world of **data science**, providing me with important capabilities that unlock its potential. My journey began with the fundamentals of **mathematics and statistics**, delving into possibility, linear algebra, calculus, and premise proof. These concepts gave me the substantial lens through which I began to see hidden patterns and stories in the data. **Python**, the versatile and successful language of data scientists, has become my tool of choice. Libraries like NumPy and Pandas have allowed me to manipulate and examine data sets with ease. Imagine changing a plain **CSV** document into a well-structured numerical table – that's the power I wielded! However the data, as I discovered, is not always flawless. **Unfinished, noisy and inconsistent** - it threw curveballs that I had to learn to deal with. This is where **data manipulation and cleaning** came into play, teaching me techniques to transform messy data into usable formats. **Data visualization** has become the art of translating numbers into stories. Charts, graphs, and dashboards became my paintbrushes, allowing me to paint compelling narratives from the data. I can now communicate complicated findings in a positive way, ensuring they resonate with both technical and non-technical audiences. This journey was one of constant learning and discovery. The more I explore, the more I realize the vastness of the data science landscape. Yet with each challenge, I gain confidence and understanding, driven by a burning desire to find the secrets hidden in the data. My path is far from over, however this activity has equipped me with essential capabilities and has ignited a passion that will guide me forward.

5 Things what i learned from the development of this activity:

- Jupyter Notebooks knowledge reinforcement.
- Basic usage of Numpy in Python.
- Do a work with English language.
- Operations with datasets.
- Data cleaning knowledge reinforcement.

University of Colima.

Faculty of Mechanical and Electrical Engineering.

Intelligent Computing Engineering.

UNIVERSIDAD DE COLIMA

Data Analysis and Visualization.

6°D.

Date: 21/02/2024.

Place: Mexico, Colima, Coquimatlan.

CRISTIAN ARMANDO LARIOS BRAVO.

In [ ]:
```
import pandas as pd
import numpy as np
```

3. Upload it to the jupyter notebook and apply the next methods:

a. df = pd.read_csv(ruta, header=None), df.head(), df.tail()

In [ ]:
```
# We load the dataset using pandas
# We use a structure named DataFrame
df = pd.read_csv('dataset.csv')
```

In [ ]:
```
# Display the name of the dataset columns
df.columns
```

Out[ ]:
```
Index(['3', '?', 'alfa-romero', 'gas', 'std', 'two', 'convertible', 'rwd',
       'front', '88.60', '168.80', '64.10', '48.80', '2548', 'dohc', 'four',
       '130', 'mpfi', '3.47', '2.68', '9.00', '111', '5000', '21', '27',
       '13495'],
      dtype='object')
```

In [ ]:
```
# Display the first five rows of the dataset
df.head()
```

Out[ ]:

	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.60	...	130	mpfi	3.47	2
0	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.6	...	130	mpfi	3.47	2
1	1	?	alfa-romero	gas	std	two	hatchback	rwd	front	94.5	...	152	mpfi	2.68	3
2	2	164	audi	gas	std	four	sedan	fwd	front	99.8	...	109	mpfi	3.19	3
3	2	164	audi	gas	std	four	sedan	4wd	front	99.4	...	136	mpfi	3.19	3
4	2	?	audi	gas	std	two	sedan	fwd	front	99.8	...	136	mpfi	3.19	3

5 rows × 26 columns

In [ ]:
```
# Display the last five rows of the dataset
df.tail()
```

Out[ ]:

	3	?	alfa-romero	gas	std	two	convertible	rwd	front	88.60	...	130	mpfi	3.
199	-1	95	volvo	gas	std	four	sedan	rwd	front	109.1	...	141	mpfi	3
200	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109.1	...	141	mpfi	3
201	-1	95	volvo	gas	std	four	sedan	rwd	front	109.1	...	173	mpfi	3
202	-1	95	volvo	diesel	turbo	four	sedan	rwd	front	109.1	...	145	idi	3
203	-1	95	volvo	gas	turbo	four	sedan	rwd	front	109.1	...	141	mpfi	3

5 rows × 26 columns

In [ ]:
```
# Display the dataset configuration: (number of rows, number of columns)
df.shape
```

Out[ ]:
```
(204, 26)
```

In [ ]:
```
# Know the general statistics of the dataset
df.describe()
```

Out[ ]:

	3	88.60	168.80	64.10	48.80	2548	130
count	204.000000	204.000000	204.000000	204.000000	204.000000	204.000000	204.000000
mean	0.823529	98.806373	174.075000	65.916667	53.749020	2555.602941	126.892157
std	1.239035	5.994144	12.362123	2.146716	2.424901	521.960820	41.744569
min	-2.000000	86.600000	141.100000	60.300000	47.800000	1488.000000	61.000000
25%	0.000000	94.500000	166.300000	64.075000	52.000000	2145.000000	97.000000
50%	1.000000	97.000000	173.200000	65.500000	54.100000	2414.000000	119.500000
75%	2.000000	102.400000	183.200000	66.900000	55.500000	2939.250000	142.000000
max	3.000000	120.900000	208.100000	72.300000	59.800000	4066.000000	326.000000

b. Notice that the dataset does not have column names. In order to solve it, create a list

structure of headers and assign it to the dataset. List the column names and add them to

the dataset, i.e.:

In [ ]:
```
encabezados = ["symboling","normalized-losses","make","fuel-type","aspiration", "nu
df.columns = encabezados
```

Replace the records in the dataset that have the value “?” for Nan's. For this purpose, use

the Pandas command: df.replace("?", np.NaN)

In [ ]:
```
df.replace("?", np.NaN)
```

Out[ ]:

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engin locatic
0	3	NaN	alfa-romero	gas	std	two	convertible	rwd	fro
1	1	NaN	alfa-romero	gas	std	two	hatchback	rwd	fro
2	2	164	audi	gas	std	four	sedan	fwd	fro
3	2	164	audi	gas	std	four	sedan	4wd	fro
4	2	NaN	audi	gas	std	two	sedan	fwd	fro
...	...	...	...	...	...	...	...	...	
199	-1	95	volvo	gas	std	four	sedan	rwd	fro
200	-1	95	volvo	gas	turbo	four	sedan	rwd	fro
201	-1	95	volvo	gas	std	four	sedan	rwd	fro
202	-1	95	volvo	diesel	turbo	four	sedan	rwd	fro
203	-1	95	volvo	gas	turbo	four	sedan	rwd	fro

204 rows × 26 columns

Convert column titles "city-mpg", "highway-mpg", to "city-kpl", "highway-kpl". Maybe you

need to add new columns, investigate the command.

In [ ]:
```
# df["city-kpl"] = df["city-mpg"] * 0.425144
df["city-kpl"] = df["city-mpg"] * 0.43
df
```

Out[ ]:

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engin locatic
0	3	?	alfa-romero	gas	std	two	convertible	rwd	fro
1	1	?	alfa-romero	gas	std	two	hatchback	rwd	fro
2	2	164	audi	gas	std	four	sedan	fwd	fro
3	2	164	audi	gas	std	four	sedan	4wd	fro
4	2	?	audi	gas	std	two	sedan	fwd	fro
...	...	...	...	...	...	...	...	...	
199	-1	95	volvo	gas	std	four	sedan	rwd	fro
200	-1	95	volvo	gas	turbo	four	sedan	rwd	fro
201	-1	95	volvo	gas	std	four	sedan	rwd	fro
202	-1	95	volvo	diesel	turbo	four	sedan	rwd	fro
203	-1	95	volvo	gas	turbo	four	sedan	rwd	fro

204 rows × 28 columns

Investigate the English to MKS measurement system conversion tables and also the Pandas

commands to convert the values in the "city-mpg" and "highway-mpg", columns from

milesper gallon to kilometers per liter. Execute on the dataset the pandas commands to

do this operation and see and validate the results.

In [ ]:
```
# df["highway-kpl"] = df["highway-mpg"] * 0.425144
df["highway-kpl"] = df["highway-mpg"] * 0.43
validate = df["highway-kpl"] == df["highway-mpg"] * 0.43
print(validate.all())
df
```

```
True
```

Out[ ]:

	symboling	normalized-losses	make	fuel-type	aspiration	num-of-doors	body-style	drive-wheels	engin locatic
0	3	?	alfa-romero	gas	std	two	convertible	rwd	fro
1	1	?	alfa-romero	gas	std	two	hatchback	rwd	fro
2	2	164	audi	gas	std	four	sedan	fwd	fro
3	2	164	audi	gas	std	four	sedan	4wd	fro
4	2	?	audi	gas	std	two	sedan	fwd	fro
...	...	...	...	...	...	...	...	...	
199	-1	95	volvo	gas	std	four	sedan	rwd	fro
200	-1	95	volvo	gas	turbo	four	sedan	rwd	fro
201	-1	95	volvo	gas	std	four	sedan	rwd	fro
202	-1	95	volvo	diesel	turbo	four	sedan	rwd	fro
203	-1	95	volvo	gas	turbo	four	sedan	rwd	fro

204 rows × 28 columns