How the development of this activity has helped me to improve my skills in the domain of dataset cleaning and exploratory analysis?

I learned from **understanding data**; Engaging in scoping and data set cleaning tasks requires a solid understanding of the data you are working with. This involves understanding the composition of the data set, the meaning of different changes, and the general environment in which the data was collected. It gave me technical competence; **Cleaning data sets** typically involves the use of **programming languages** such as **Python** or **R**, as well as specialized libraries such as **Pandas**. Through practice, I become more proficient in using these tools and learn different functionalities and procedures for **data manipulation and investigation**. This technical proficiency not only allows me to clean data sets more efficiently, but also lays a solid foundation for more **advanced data study tasks**. It gave me problem-solving skills; Cleaning data sets is rarely easy, because data sets often contain errors, inconsistencies, and missing values that need to be addressed. Participating in these activities helps me develop problem-solving skills while devising tactics to handle various **data problems**. I learn to think critically and creatively to find resolutions that guarantee the integrity and quality of the data. It gave me statistical and analytical skills; Exploratory study involves summarizing and **visualizing data to find patterns, trends, and interrelationships**. By performing exploratory study exercises, I improve my **statistical and analytical skills**, learning to choose appropriate summary statistics, produce informative observations, and interpret my findings accurately. I finally got attention to detail; Cleaning the data set requires a high degree of attention to detail to detect and **address errors or inconsistencies** in a positive way.

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

- Jupyter Notebooks knowledge reinforcement.
- Basic usage of Pandas in Python
- Do a work with English language.
- How works an dataframe in Python.
- What is EDA? an introduction to Data Analysis.

University of Colima.

Faculty of Mechanical and Electrical Engineering.

Intelligent Computing Engineering.

UNIVERSIDAD DE COLIMA

Data Analysis and Visualization.

6°D.

Date: 20/02/2024.

Place: Mexico, Colima, Villa de Alvarez.

CRISTIAN ARMANDO LARIOS BRAVO.

S3-P1-U1-AC-O3 Exploratory Analysis of the Titanic Dataset

🥑 INTRODUCTION

Exploratory Data Analysis (EDA), also known as Data Exploration, is a step in the Data Analysis Process, where a number of techniques are used to better understand the dataset being used. 'Understanding the dataset' can refer to a number of things including but not limited to:

- Extracting important variables and leaving behind useless variables.

- Identifying outliers, missing values, or human error.
- Understanding the relationship(s), or lack of, between variables.
- Ultimately, maximizing your insights of a dataset and minimizing potential error that may occur later in the process.

On the other hand, Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series. It is free software released under the three-clause BSD license.

🎯 MAIN GOAL OF THIS ACTIVITY

The main objective of this activity is for you to learn how to apply the data exploratory analysis process to the Titanic dataset.

🐇 DEVELOPMENT

1. Create a Jupyter Notebook y name it S3-P1-U1-AC-03 "Add Your Full Name"
2. Localize and Load the Titanic dataset located at https://www.kaggle.com/ or use the dataset file attached in this message
3. To know the dataset characteristic , apply to the dataset the Pandas methods seen in class :

- df = pd.read_csv(ruta, header=None)
- df.head(), df.tail(), df.shape()
- df.describe()
- df. columns.values
- df.dtypes

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

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

Exploratory analysis

The first step after loading the dataset into a **Pandas** DataFrame is to perform the **exploratory analysis**

- It consists of using a series of **Pandas** methods to obtain the characteristics of the dataset.

In [ ]:
```
# Let's display the characteristics of the dataset
df
```

Out[ ]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	
0	892	0	3	Kelly, Mr. James	male	34.5	0	0	330911	
1	893	1	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	
2	894	0	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	
3	895	0	3	Wirz, Mr. Albert	male	27.0	0	0	315154	
4	896	1	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	1
...	...	...	...	...	...	...	...	...	...	
413	1305	0	3	Spector, Mr. Woolf	male	NaN	0	0	A.5. 3236	
414	1306	1	1	Oliva y Ocana, Dona. Fermina	female	39.0	0	0	PC 17758	10
415	1307	0	3	Saether, Mr. Simon Sivertsen	male	38.5	0	0	SOTON/O.Q. 3101262	
416	1308	0	3	Ware, Mr. Frederick	male	NaN	0	0	359309	
417	1309	0	3	Peter, Master. Michael J	male	NaN	1	1	2668	2

418 rows × 12 columns

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

Out[ ]:
```
Index(['PassengerId', 'Survived', 'Pclass', 'Name', 'Sex', 'Age', 'SibSp',
       'Parch', 'Ticket', 'Fare', 'Cabin', 'Embarked'],
      dtype='object')
```

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

Out[ ]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare
0	892	0	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292
1	893	1	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000
2	894	0	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875
3	895	0	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625
4	896	1	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875

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

Out[ ]:

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	
413	1305	0	3	Spector, Mr. Woolf	male	NaN	0	0	A.5. 3236	8
414	1306	1	1	Oliva y Ocana, Dona. Fermina	female	39.0	0	0	PC 17758	108
415	1307	0	3	Saether, Mr. Simon Sivertsen	male	38.5	0	0	SOTON/O.Q. 3101262	7
416	1308	0	3	Ware, Mr. Frederick	male	NaN	0	0	359309	8
417	1309	0	3	Peter, Master. Michael J	male	NaN	1	1	2668	22

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

Out[ ]:
```
(418, 12)
```

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

Out[ ]:

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	418.000000	418.000000	418.000000	332.000000	418.000000	418.000000	417.000000
mean	1100.500000	0.363636	2.265550	30.272590	0.447368	0.392344	35.627188
std	120.810458	0.481622	0.841838	14.181209	0.896760	0.981429	55.907576
min	892.000000	0.000000	1.000000	0.170000	0.000000	0.000000	0.000000
25%	996.250000	0.000000	1.000000	21.000000	0.000000	0.000000	7.895800
50%	1100.500000	0.000000	3.000000	27.000000	0.000000	0.000000	14.454200
75%	1204.750000	1.000000	3.000000	39.000000	1.000000	0.000000	31.500000
max	1309.000000	1.000000	3.000000	76.000000	8.000000	9.000000	512.329200

In [ ]:
```
# Imagine that they ask us for the number of survivors of the Titanic
# reported by this dataset
```

```
print("Number of survivors: {}".format(df["Survived"].value_counts()))
```

```
Number of survivors: Survived
0    266
1    152
Name: count, dtype: int64
```

In [ ]:
```
# Datatypes of the dataset
print(df.dtypes)
```

```
PassengerId      int64
Survived         int64
Pclass           int64
Name            object
Sex             object
Age            float64
SibSp            int64
Parch            int64
Ticket          object
Fare           float64
Cabin           object
Embarked        object
dtype: object
```

4. Counts the number of surviving people classified as women and men.

In [ ]:
```
womenS = df[(df["Sex"]=="female") & (df["Survived"]==1)].shape[0]
menS = df[(df["Sex"]=="male") & (df["Survived"]==1)].shape[0]
print(f"Number of surviving people:\n\
Men: {menS}\n\
Women: {womenS}")
```

```
Number of surviving people:
Men: 0
Women: 152
```

5. Counts the number of women over 30 years old who survived the shipwreck.

In [ ]:
```
women30S = df[(df["Age"]>30) & (df["Survived"]==1) & (df["Sex"]=="female")].shape[0
print(f"Number of women over 30 years old who survived the shipwrek: {women30S}")
```

```
Number of women over 30 years old who survived the shipwrek: 50
```

6. Count the number of crew members over 21 years of age.

In [ ]:
```
crew = df[(df["Pclass"]==3) & (df["Age"]>21)].shape[0]
print(f"Number of crew members over 21 years old: {crew}")
```

```
Number of crew members over 21 years old: 91
```