 Experiment: - 1

## Student Name: Sahul Kumar Parida UID: 20BCS4919

**Branch: CSE Section/Group: WMA-904/B**

**Semester: 5th Subject Code: 20CSP-317**

**Subject Name: MACHINE LEARNING LAB**

**Aim/Overview of the practical:**

Exploratory Data Analysis on any data set.

# Task to be done:

To perform Data Analysis on any standard dataset.

# Apparatus/Simulator used:

* Jupyter Notebook/Google Collab
* Python
* pandas Library



**Theory:**

Pandas is a Python library used for working with data sets. It is use for analyzing, cleaning, exploring, and manipulating data. It allows us to analyze big data and make conclusions based on statistical theories. It can clean messy data sets, and make them readable and relevant.

Some dataframes of pandas used in this practical are:

1. dtype

Return the dtypes in the DataFrame. This returns a Series with the data type of each column. The result’s index is the original DataFrame’s columns. Columns with mixed types are stored with the object.

1. axes

Return a list representing the axes of the DataFrame. It has the row axis labels and column axis labels as the only members. They are returned in that order.

1. size

Return an int representing the number of elements in this object. Return the number of rows if Series. Otherwise return the number of rows times number of columns if DataFrame.

1. shape

Return a tuple representing the dimensionality of the DataFrame.

1. empty

Indicator whether Series/DataFrame is empty. True if Series/DataFrame is entirely empty (no items), meaning any of the axes are of length 0.

1. head

Return the first n rows. This function returns the first n rows for the object based on position. It is useful for quickly testing if your object has the right type of data in it.

1. count

Count non-NA cells for each column or row.

1. tail

Return the last n rows. This function returns last n rows from the object based on position. It is useful for quickly verifying data, for example, after sorting or appending rows.

1. sort\_values

Sort by the values along either axis.

1. dropna

Remove missing values.

1. drop\_duplicates

Return DataFrame with duplicate rows removed.

# Code and Output:











