Name: Shashwati Kale

Class: CSE-AI SY B

PRN: 22310704

Roll no.: 282025

**Assignment 1**

# Problem Statement

Perform the following operations using R/Python on suitable data sets: a) read data from different formats (like csv, xls)

1. Find Shape of Data
2. Find Missing Values
3. Find data type of each column
4. Finding out Zero's
5. Indexing and selecting data, sort data,
6. Describe attributes of data, checking data types of each column,
7. counting unique values of data, format of each column, converting variable data type (e.g. from long to short, vice versa)

# Objective

The objective of this assignment is to perform fundamental data exploration operations on a dataset using Python. These operations include reading data from different file formats, analyzing the dataset’s structure, handling missing values, checking data types, identifying zero values, and performing basic data manipulation tasks.

# Resources Used

Programming Language: Python

Libraries Used: pandas – for data manipulation and analysis numpy – for numerical operations

matplotlib / seaborn (optional) – for data visualization

Dataset: heart.csv (heart disease dataset)

# Introduction to Pandas

Pandas is an essential Python library used for data manipulation and analysis. It provides two primary data structures:

Series: A one-dimensional labeled array

DataFrame: A two-dimensional labeled table with rows and columns

Pandas makes it easy to handle structured data, offering functionalities for data reading, cleaning, transformation, and analysis.

# Basic Functions Used in the Program

Here are some essential functions utilized in this assignment:

pd.read\_csv('file.csv') – Reads a CSV file into a DataFrame pd.read\_excel('file.xlsx') – Reads an Excel file df.shape – Returns the number of rows and columns df.isnull().sum() – Identifies missing values df.dtypes – Displays data types of each column df[df == 0].count() – Counts zero values in the dataset df.sort\_values(by='column\_name') – Sorts the dataset based on a column df.describe() – Provides a statistical summary of the dataset df.nunique() – Counts unique values in each column

df['column\_name'].astype('int') – Converts the data type of a column

# Methodology of Solution

## Data Collection

Imported a dataset in various formats (CSV, XLS)

Used pandas.read\_csv() and pandas.read\_excel() to load the data into a DataFrame

## Data Exploration

Finding Shape of Data: Used df.shape to get the number of rows and columns

Checking Data Types: Used df.dtypes to determine the data type of each column

## Handling Missing Values

Used df.isnull().sum() to count missing values

Handled missing values by removing them (df.dropna()) or filling them with mean/median (df.fillna(value))

Finding Zero Values

Used df[df == 0].count() to check how many zero values exist in each column

## Data Selection and Indexing

Used df.loc[] and df.iloc[] for row and column selection

Selected specific rows and columns based on conditions

Sorting Data

Used df.sort\_values(by='column\_name') to arrange data in ascending/descending order

Attribute Description

Used df.describe() to generate statistical insights like mean, median, and standard deviation

Counting Unique Values

Used df.nunique() to determine unique values per column

Data Type Conversion

Used df['column\_name'].astype(new\_dtype) to change data types when necessary

# Advantages

Efficient Data Handling: Pandas makes data manipulation faster and easier

Comprehensive Data Analysis: Functions like describe() provide instant statistical insights Flexibility: Supports multiple file formats (CSV, Excel, JSON, etc.)

Missing Data Management: Offers multiple ways to handle missing values

# Disadvantages

Memory Consumption: Large datasets may require significant memory

Performance Bottlenecks: Operations like sorting and filtering can be slow on massive datasets

Dependency on Libraries: Pandas relies on additional libraries like NumPy, which may require installation and updates

# Conclusion

This assignment covered fundamental data exploration techniques using Python. We successfully loaded datasets from different formats, checked their structure, handled missing values, sorted data, and analyzed key attributes. These operations are crucial in preprocessing, ensuring clean and well-structured data for further machine learning tasks.