**Mastering Pandas and NumPy for Data Analysis**

This project focuses on data preprocessing, manipulation, and analysis using Pandas and NumPy in Python. Below are the key steps involved:

**Step 1: Introduction**

* Pandas is a powerful Python library designed for efficient data handling and analysis.
* NumPy supports high-performance numerical computations and forms the core of Pandas operations.

**Step 2: Installation and Importing Libraries**

* Install Pandas using: pip install pandas
* Install NumPy using: pip install numpy
* Import them as follows:

python

import pandas as pd

import numpy as np

**Step 3: Data Loading (Ingestion)**

* Load datasets from CSV files using pd.read\_csv().

**Step 4: Data Exploration**

* View initial records with .head().
* Get descriptive statistics using .describe().
* Check data types and missing values with .info().

**Step 5: Data Cleaning & Handling Missing Values**

* Identify missing values using .isnull().sum().
* Remove missing values using .dropna().
* Fill missing values with a specific value using .fillna(value, inplace=True).
* Eliminate duplicate records using .drop\_duplicates().

**Step 6: Data Selection and Filtering**

* Extract specific columns: df[['column1', 'column2']].
* Apply conditional filtering: df[df['column'] > value].
* Use indexing methods like .iloc[] (position-based) and .loc[] (label-based).

**Step 7: Data Transformation**

* Create new columns: df['new\_col'] = df['col1'] \* 10.
* Remove columns using df.drop('column\_name', axis=1).
* Rename columns using df.rename(columns={'old\_name': 'new\_name'}).

**Step 8: Sorting and Aggregation**

* Sort data using .sort\_values(by='column', ascending=False).
* Aggregate data using .groupby('column').agg({'col': 'mean'}).

**Step 9: Indexing Operations**

* Set multiple indexes using .set\_index(['col1', 'col2']).
* Reset index using .reset\_index().

**Step 10: Data Analysis & Insights**

* Count unique values using .value\_counts().
* Compute key statistical metrics like mean, median, and standard deviation.

**Conclusion**

This project provides a structured approach to data preprocessing, manipulation, and analysis using Pandas and NumPy.