Task 5 Report
Time Series Analysis with Pandas

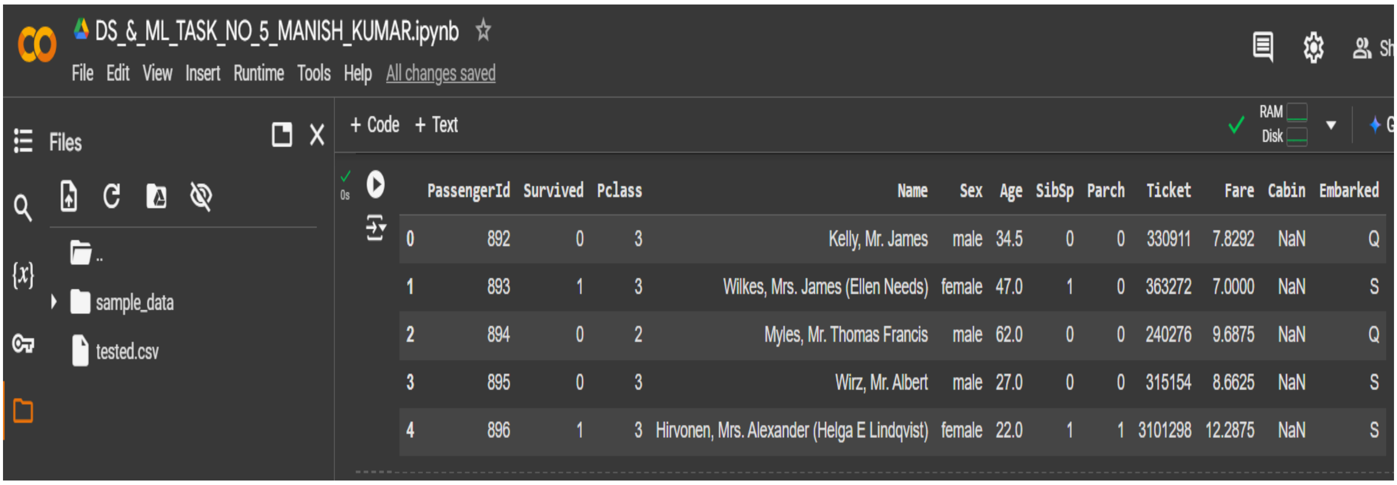
1. Task Description

This task focuses on performing **time series analysis** using Pandas. The key objectives include:

- Simulating a datetime column for time-based indexing and operations.
- Handling missing values in numeric columns.
- Resampling the dataset to calculate weekly averages.
- Computing rolling statistics like rolling mean and rolling standard deviation.
- Creating lagged features to identify time-based dependencies.
- Visualizing the time series using Matplotlib to understand trends, patterns, and variability.

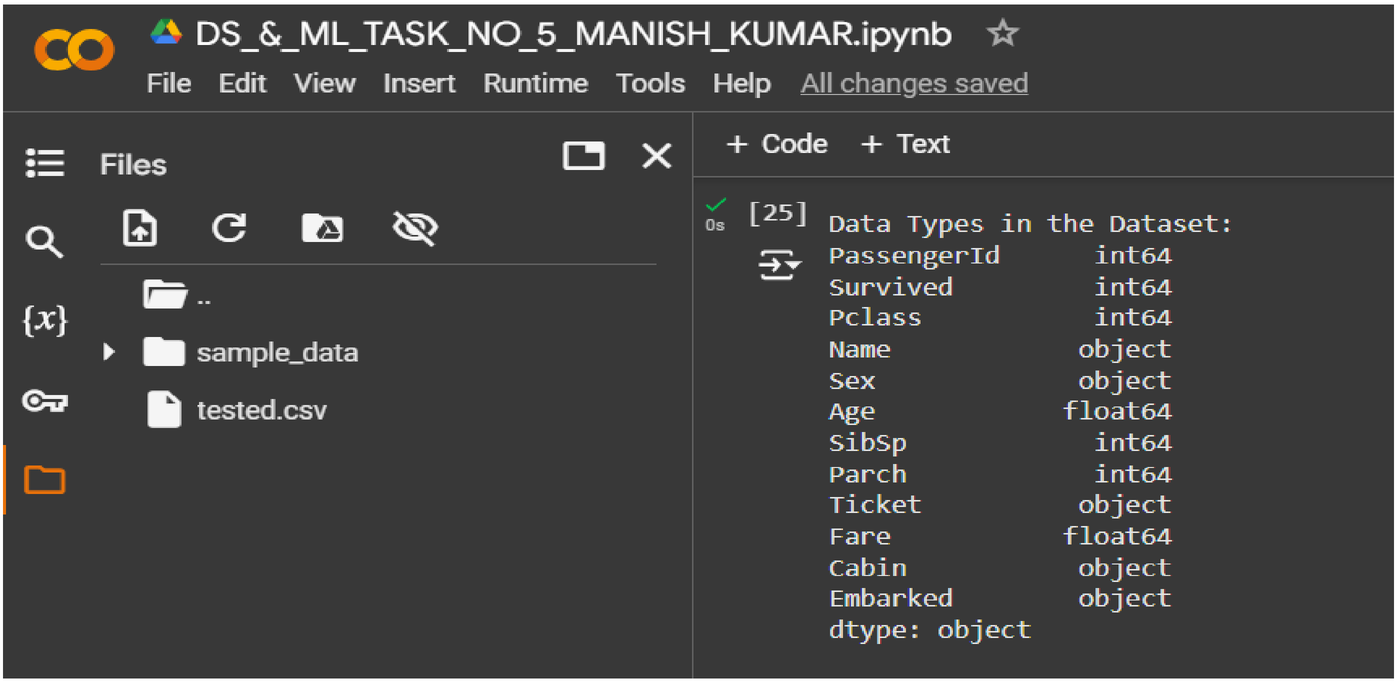
2. Attach Screenshot of Output

- **Dataset Preview:**


DS_&_ML_TASK_NO_5_MANISH_KUMAR.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Files
+ Code + Text
RAM
Disk
sample_data
tested.csv
0s

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

- **Data Types in the Dataset:**

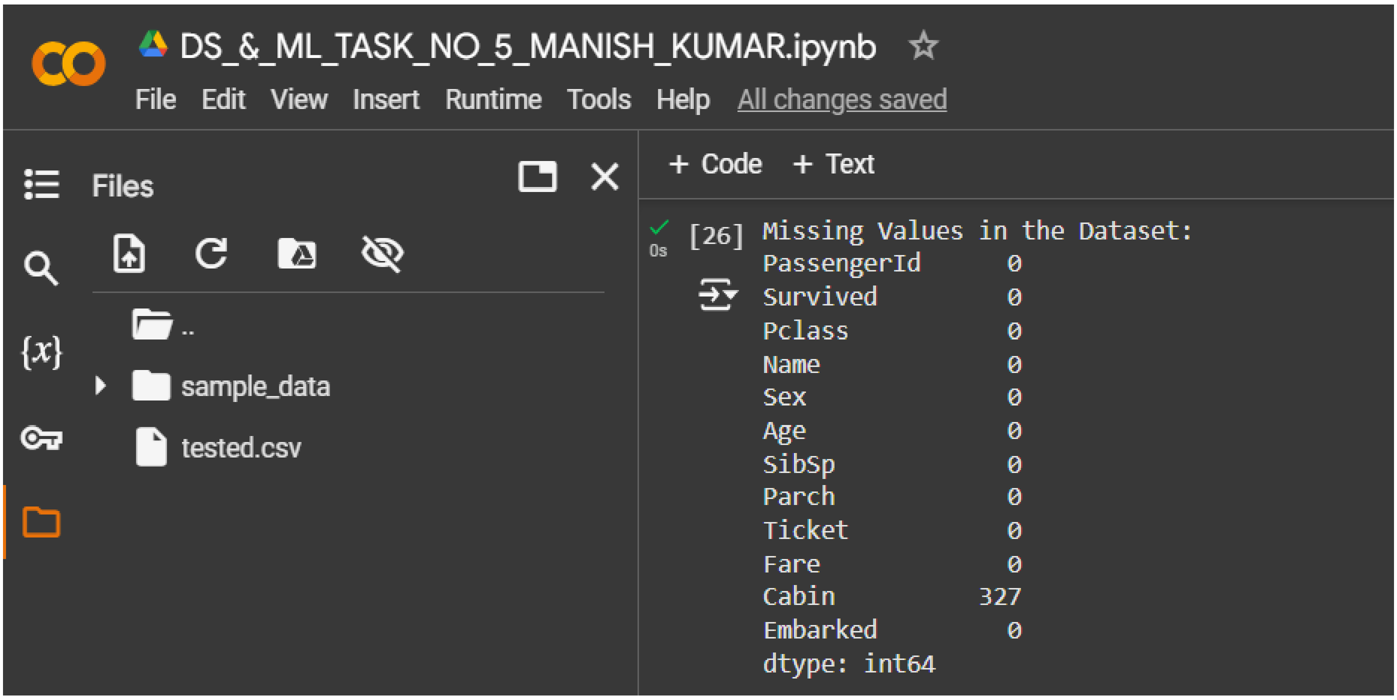

DS_&_ML_TASK_NO_5_MANISH_KUMAR.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Files
+ Code + Text
sample_data
tested.csv
0s
[25]

```
Data Types in the Dataset:
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
```

Task 5 Report

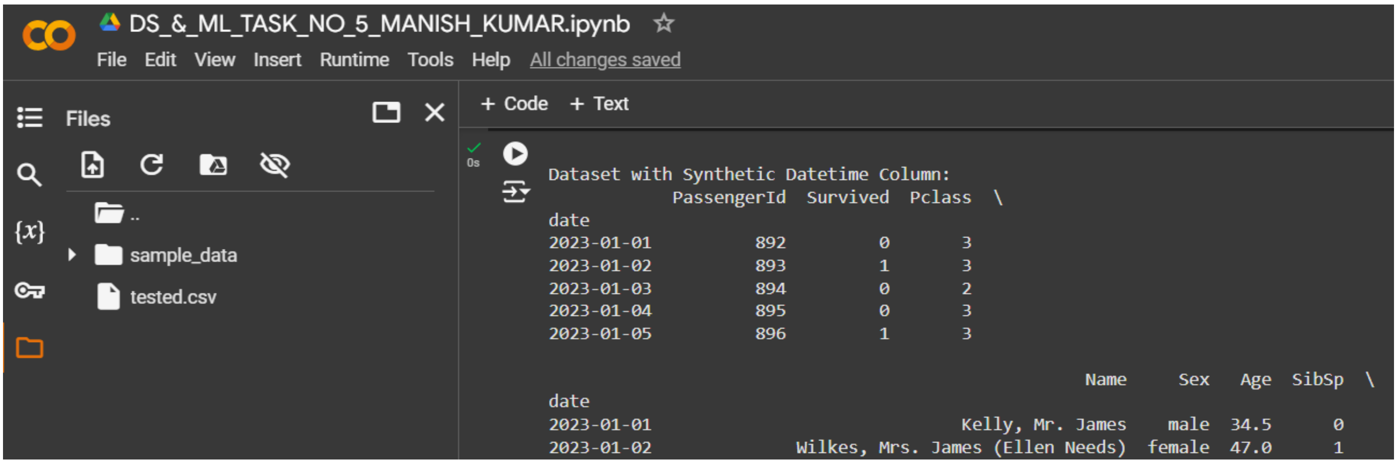
Time Series Analysis with Pandas

❖ Missing values in the Dataset:


DS_&_ML_TASK_NO_5_MANISH_KUMAR.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Files
..
sample_data
tested.csv
+ Code + Text
0s
[26]

```
Missing Values in the Dataset:
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age              0
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          327
Embarked         0
dtype: int64
```

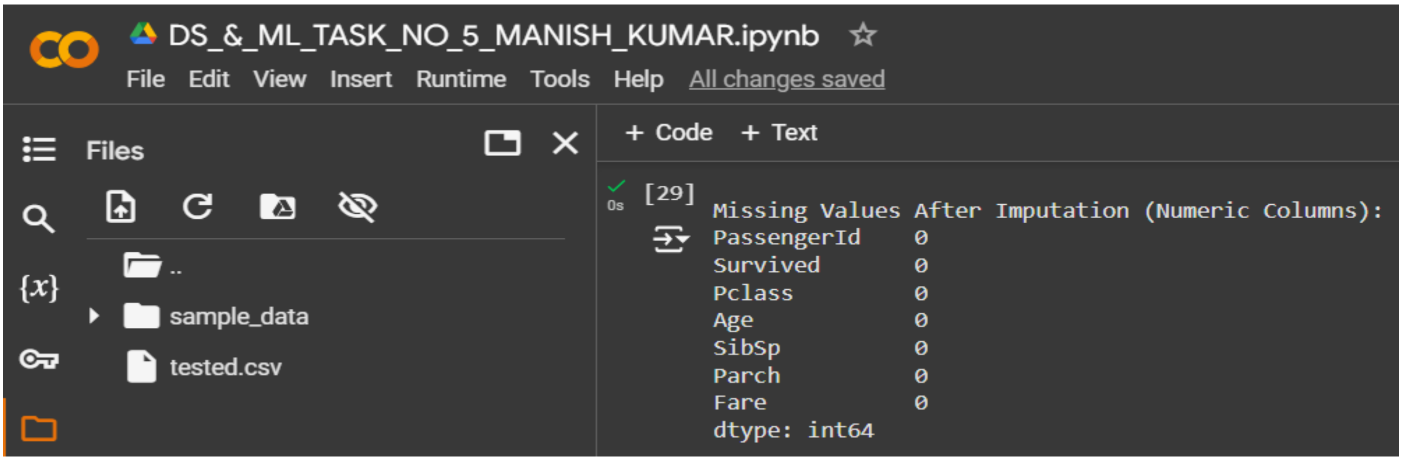
❖ Dataset with Synthetic Datetime Column:


DS_&_ML_TASK_NO_5_MANISH_KUMAR.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Files
..
sample_data
tested.csv
+ Code + Text
0s

```
Dataset with Synthetic Datetime Column:
            PassengerId  Survived  Pclass  \
date
2023-01-01          892         0       3
2023-01-02          893         1       3
2023-01-03          894         0       2
2023-01-04          895         0       3
2023-01-05          896         1       3

                                                    Name     Sex   Age  SibSp  \
date
2023-01-01                              Kelly, Mr. James    male  34.5      0
2023-01-02              Wilkes, Mrs. James (Ellen Needs)  female  47.0      1
```

❖ Missing values after Imputation:

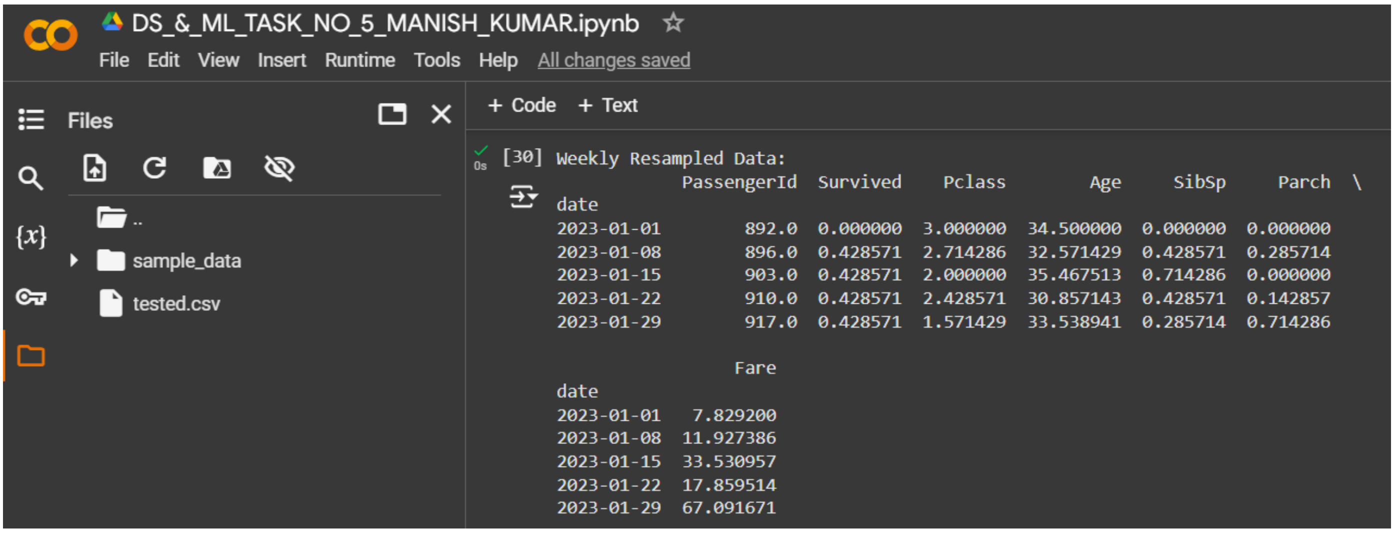

DS_&_ML_TASK_NO_5_MANISH_KUMAR.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Files
..
sample_data
tested.csv
+ Code + Text
0s
[29]

```
Missing Values After Imputation (Numeric Columns):
PassengerId    0
Survived       0
Pclass         0
Age            0
SibSp          0
Parch          0
Fare           0
dtype: int64
```

Task 5 Report

Time Series Analysis with Pandas

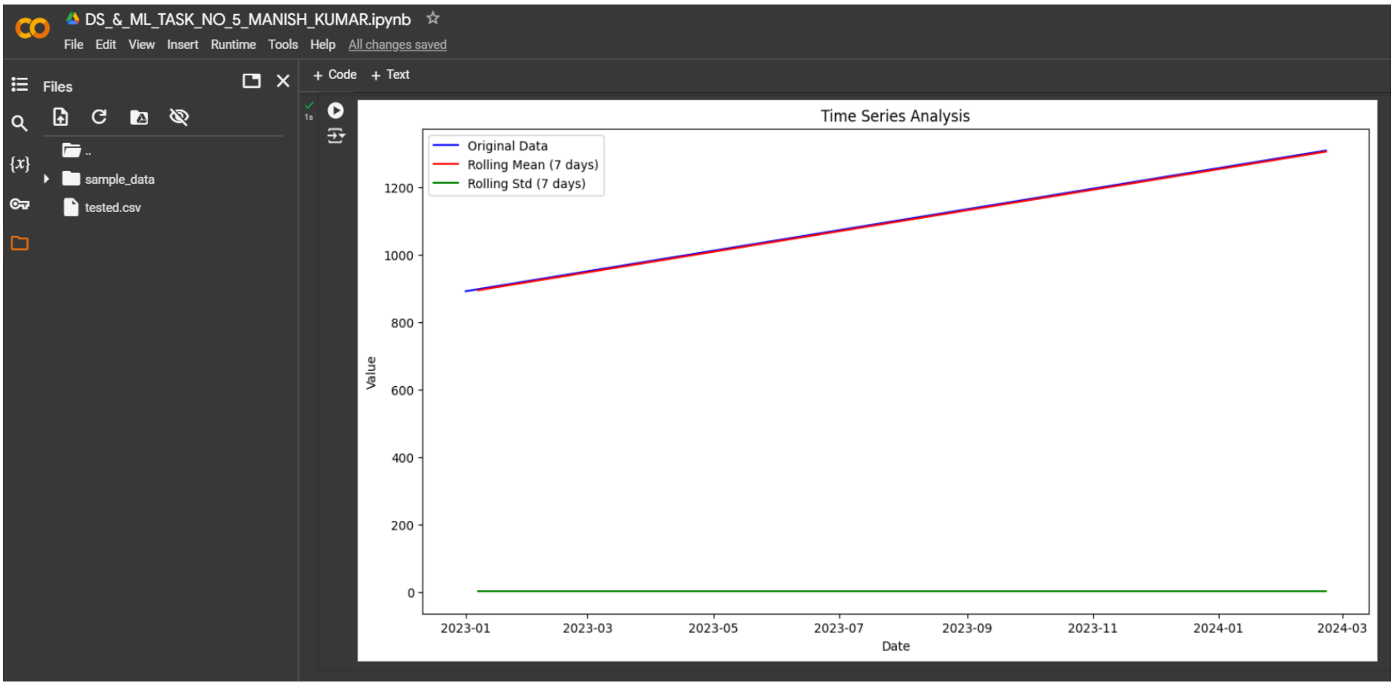
❖ Weekly Resampled Data:


DS_&_ML_TASK_NO_5_MANISH_KUMAR.ipynb
File Edit View Insert Runtime Tools Help All changes saved
Files
+ Code + Text
..
sample_data
tested.csv

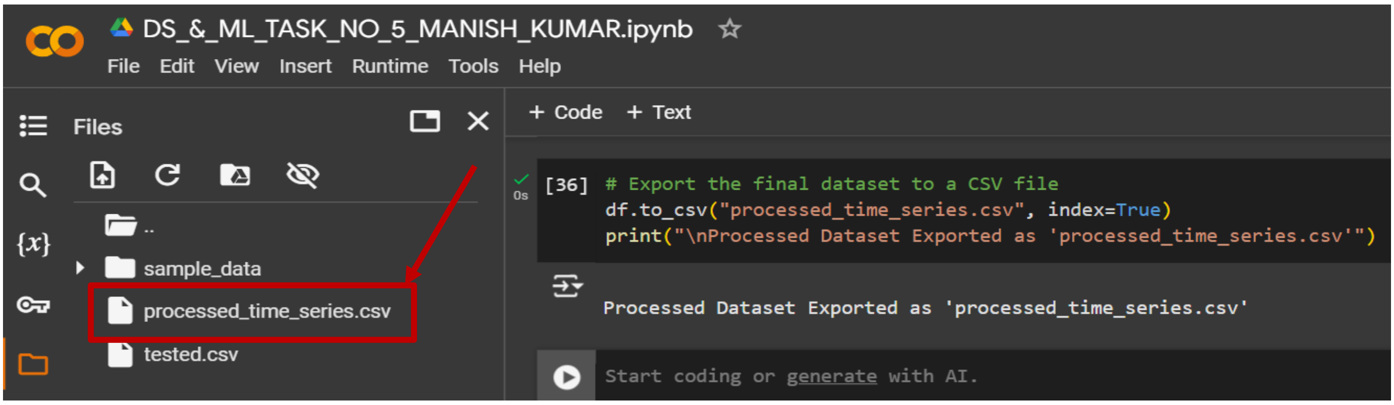
```
[30] Weekly Resampled Data:
            PassengerId  Survived    Pclass        Age     SibSp     Parch  \
date
2023-01-01        892.0  0.000000  3.000000  34.500000  0.000000  0.000000
2023-01-08        896.0  0.428571  2.714286  32.571429  0.428571  0.285714
2023-01-15        903.0  0.428571  2.000000  35.467513  0.714286  0.000000
2023-01-22        910.0  0.428571  2.428571  30.857143  0.428571  0.142857
2023-01-29        917.0  0.428571  1.571429  33.538941  0.285714  0.714286

                 Fare
date
2023-01-01   7.829200
2023-01-08  11.927386
2023-01-15  33.530957
2023-01-22  17.859514
2023-01-29  67.091671
```

❖ Time Series Analysis:



❖ Processed Dataset Exported as 'processed_time_series.csv':


DS_&_ML_TASK_NO_5_MANISH_KUMAR.ipynb
File Edit View Insert Runtime Tools Help
Files
+ Code + Text
..
sample_data
processed_time_series.csv
tested.csv
Start coding or generate with AI.

```
[36] # Export the final dataset to a CSV file
     df.to_csv("processed_time_series.csv", index=True)
     print("\nProcessed Dataset Exported as 'processed_time_series.csv'")
```

```
Processed Dataset Exported as 'processed_time_series.csv'
```

Task 5 Report

Time Series Analysis with Pandas

3. Describe Widget/Algorithm Used in Task

Algorithm Used:

The task leverages the following algorithms and techniques:

- **Data Preprocessing**:
  - **Datetime Simulation**: A synthetic datetime column is added to mimic time-series data, and the column is set as the index to enable time-based operations.
  - **Handling Missing Values**: Missing values in numeric columns are replaced with the column mean for smoother analysis.
- **Resampling**:
  - The numeric data is resampled to a weekly frequency using Pandas' resample method, calculating weekly averages to understand long-term trends.
- **Rolling Statistics**:
  - **Rolling Mean**: A 7-day rolling window is applied to calculate the average value over the last 7 days, smoothing short-term fluctuations.
  - **Rolling Standard Deviation**: A 7-day rolling window is applied to measure variability in the data.
- **Lagged Features**:
  - **Lagging**: Shifted versions of the data are created (e.g., lagged by 1 day or 7 days) to capture time-dependent patterns.
- **Visualization**:
  - **Line Plots**: Rolling mean, rolling standard deviation, and the original data are plotted to visually compare trends and variability over time.

Tools Used:

- **Pandas**:

  \

  - For data manipulation, resampling, rolling statistics, and handling missing values.
  - Used the resample, rolling, and shift functions to perform time-series-specific operations.
- **NumPy**:
  - Assisted in handling numerical operations while imputing missing values.
- **Matplotlib**:
  - For creating visualizations such as line plots to showcase trends, rolling statistics, and lagged features.
- **Jupyter/Colab Environment**:
  - Provided an interactive Python environment for implementing the time-series analysis and visualizing results.

***** The End *****