EX -1
23/01/2025

IMPLEMENTING THE PROGRAM FOR THE TIME SERIES DATA CLEANING AND PREPROCESSING TECHNIQUES

AIM :

To implement the program for the Time Series Data Cleaning And Preprocessing Techniques.

Procedure and Code :

Step 1 - Import the Files and Libraries .

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

Step 2 - Describe and Read the Data

```
df.head(10)
```

	Row	Day	Day.Of.Week	Date	Page.Loads	Unique.Visits	First.Time.Visits	Returning.Visits
0	1	Sunday	1	9/14/2014	2,146	1,582	1,430	152
1	2	Monday	2	9/15/2014	3,621	2,528	2,297	231
2	3	Tuesday	3	9/16/2014	3,698	2,630	2,352	278
3	4	Wednesday	4	9/17/2014	3,667	2,614	2,327	287
4	5	Thursday	5	9/18/2014	3,316	2,366	2,130	236
5	6	Friday	6	9/19/2014	2,815	1,863	1,622	241
6	7	Saturday	7	9/20/2014	1,658	1,118	985	133
7	8	Sunday	1	9/21/2014	2,288	1,656	1,481	175
8	9	Monday	2	9/22/2014	3,638	2,586	2,312	274
9	10	Tuesday	3	9/23/2014	4,462	3,257	2,989	268

```
df.shape
```

(2167, 8)

EX -1
23/01/2025

```
df.describe(include='all').T

df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2167 entries, 0 to 2166
Data columns (total 8 columns):
 #   Column           Non-Null Count  Dtype 
---  ------           --------------  ----- 
 0   Row              2167 non-null   int64 
 1   Day              2167 non-null   object
 2   Day.Of.Week      2167 non-null   int64 
 3   Date             2167 non-null   object
 4   Page.Loads       2167 non-null   object
 5   Unique.Visits    2167 non-null   object
 6   First.Time.Visits  2167 non-null   object
 7   Returning.Visits   2167 non-null   object
dtypes: int64(2), object(6)
memory usage: 135.6+ KB
```

Step 3 - Cleaning and preprocessing the data

```
data_null = df.notnull().sum

df['Page.Loads'] = df['Page.Loads'].str.replace(',', '').astype(int)

daywise_data = df.groupby('Day')['Page.Loads'].sum()
```

EX -1
23/01/2025

Step 4 - Dropping the duplicate and missing values

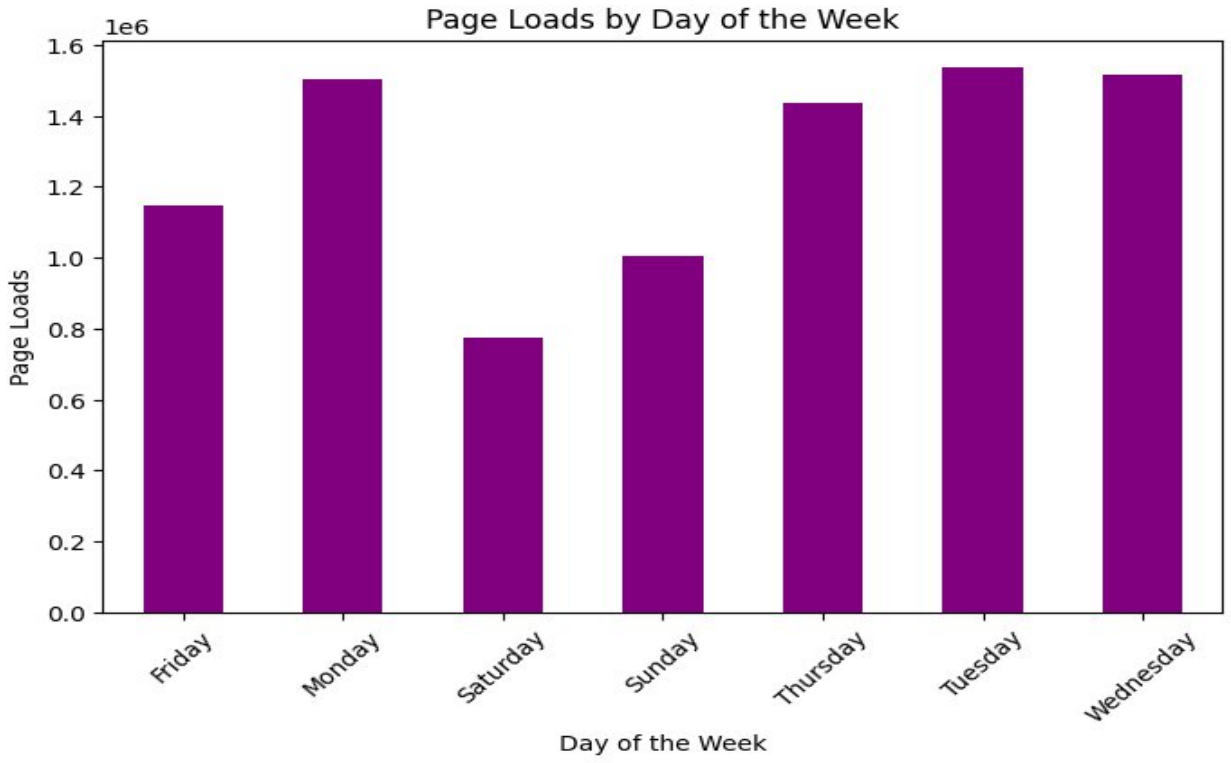
```
data = df.drop_duplicates()


print(f"Dataset now has {data.shape[0]} rows and {data.shape[1]} columns.")
```

Step 5 - visualizing the Dataset

```
daywise_data.plot(kind='bar', figsize=(8, 5), color='purple')
plt.title('Page Loads by Day of the Week')
plt.xlabel('Day of the Week')
plt.ylabel('Page Loads')
plt.xticks(rotation=45)

plt.show()
```



Result:

Thus the Program has been Executed Successfully.

EX -1
23/01/2025