PUBLIC TRANSPORTATION ANALYSIS :

CONTENT :

**Public transport analysis assesses aspects like ridership, route efficiency, and user satisfaction. By studying this data, authorities can enhance services, improve efficiency, and create sustainable urban transport systems, benefiting both commuters and the environment.**

**ENVIRONMENT SETTING :**

**Make sure you have the necessary tools and libraries installed. You can use platforms like GoogleColab, Jupyter Notebook or your local Python environment. Import the necessary libraries that aregood at manipulating data (mainly pandas).**

DATASET :

<https://www.kaggle.com/datasets/rednivrug/unisys?select=20140711.CSV>

DOWNLOAD DATASET:

**Load dataset into DataFrame using pandas library.**

DATA EXPLORATION:

**It is important to understand your data before preprocessing. You can explore your dataset using**

**various pandas functions.**

**→ This initial exploration will help you understand the structure of your data, identify missing values,**

**and identify columns that need preprocessing**

DATA PREPROCESSING :

**Data preprocessing plays a crucial role in public transport analysis by enhancing the quality and reliability of the collected data.**

**In the realm of public transport, vast amounts of data are generated daily, including passenger counts, vehicle locations, and ticket sales.**

**Preprocessing techniques involve cleaning the data to remove inconsistencies, handling missing values, and standardizing formats to ensure uniformity.**

**Additionally, preprocessing methods such as data normalization and feature scaling help in preparing the data for analysis, enabling transportation experts to gain valuable insights into passenger patterns, optimize routes, and improve overall public transport efficiency.**

DATA EXPORT:

**If you want to save your preprocessed dataset for future use, you can export it to a CSV file .**

FURTHER ANALYSIS:

**→ Now that we have a clean, pre-processed data set, we can proceed with our analysis. You can use a**

**variety of statistical and machine learning techniques to gain insights and predictions based on your**

**data.**

**→ The process described here is a basic and common approach to loading and preprocessing datasets**

**for data analysis.**

**PROGRAM:**

**# Importing necessary libraries**

**import pandas as pd**

**import matplotlib.pyplot as plt**

**import seaborn as sns**

**# Load the dataset (assuming it's in CSV format)**

**data\_url = 'URL\_TO\_YOUR\_DATASET.csv'**

**df = pd.read\_csv(data\_url)**

**# Display the first few rows of the dataset**

**print("Sample Data:")**

**print(df.head())**

**# Summary statistics of numerical columns**

**print("\nSummary Statistics:")**

**print(df[['NumberOfBoardings']].describe())**

**# Visualization: Bar chart for NumberOfBoardings vs StopName**

**plt.figure(figsize=(12, 6))**

**sns.barplot(x='StopName', y='NumberOfBoardings', data=df)**

**plt.title('Number of Boardings at Each Stop')**

**plt.xticks(rotation=90)**

**plt.xlabel('Stop Names')**

**plt.ylabel('Number of Boardings')**

**plt.show()**

**# Visualization: Line chart for NumberOfBoardings trend over Weeks**

**plt.figure(figsize=(12, 6))**

**sns.lineplot(x='WeekBeginning', y='NumberOfBoardings', data=df)**

**plt.title('Number of Boardings Trend Over Weeks')**

**plt.xticks(rotation=45)**

**plt.xlabel('Week Beginning')**

**plt.ylabel('Number of Boardings')**

**plt.show()**

OUTPUT:

