# Step 1: Import Libraries and Necessary Installation

Text, application, letter

Description automatically generatedLoad the necessary libraries and do required installation.   
Text

Description automatically generated

**Step 2: Read file.**

Loaded the bike sharing dataset file named “london\_merged.csv”. Checked memory usage using the info method in Pandas.

**Step 3: Exploratory Data Analysis (EDA)**

Analyzed the shape, feature size and statistics for the bike sharing dataset. Converted the timestamp column to actual timestamp datatype because as per the requirement the dataset needs to be sorted based on time slots from old to new before splitting into test and train. Checked if there are any categorical variables in the dataset so that we can process them accordingly during Feature Engineering. We checked for duplicates and Null values. There were no duplicates or Null values found. Found correlation between the attributes using Correlation Matrix. Real temperature in C (“t1”) and Feels like temperature in C (“t2”) are highly positively correlated. The count of new bike shares (“Cnt”) and Humidity in percentage (“hum”) are negatively correlated. Real temperature in C (“t1”) and Humidity in percentage (“hum”) are negatively correlated. Feels like temperature in C (“t2”) and Humidity in percentage (“hum”) are negatively correlated.

**Step 4: Feature Engineering**

Sorted the dataset based on timestamp in ascending order. Applied assert\_true (a unit test library function in Python used in unit testing to compare test value with true) to check and confirm whether the columns “timestamp”, “isholiday” and “isweekend” are present. Created a derived column based on “is\_holiday” and “is\_weekend” to find out a working-day/non-working day pattern with respect to the bike sharing counts. Also, the timestamp column has been disintegrated to month, year, hour, and night\_hour to learn about more hidden patterns (if any) in the dataset. This will enable the machine learning algorithms to easily learn the underlying patterns till the lowest level of granularity in the timestamp column.

**Step 5: Split the dataset into train and test.**

Split the dataset (df1) into train and test, i.e 80% of the dataset is used for training the model and 20% of the dataset is used for testing the model. Then, separated the output (Y) variable (‘cnt’) from input (X) variable for both train and test datasets.

**Step 6: Model Selection**

We checked the RMSLE scores with parameter greater\_is\_better=False for some ML models.

Table

Description automatically generated