**Advanced Mathematical Statistics (522 -03)**

**Project Report**

**Weather Prediction using ML Algorithms**

**Jayanth Mekala (02173552)**

**Anusha Musirika (02199061)**

**Thatikonda Preetham (02182487)**

**Sarvagna Kotte(02185847)**

**Roshan Pampari (02143843)**

**Problem Definition**

You want to predict a specific weather condition (e.g., daily temperature or precipitation) for a particular location or region based on historical weather data. The dataset will typically contain weather observations for multiple years, including temperature, wind speed, humidity, and other atmospheric conditions.

**About Dataset**

We have obtained our dataset from kaggle

https://www.kaggle.com/datasets/muthuj7/weather-dataset

Our Dataset contains 96453 instances and 12 distinct features as follows:

**Formatted Date**: This represents the date and time when the weather data was recorded, with time zone information included.

**Summary**: A brief description of the general weather condition (e.g., "Partly Cloudy," "Mostly Cloudy").

**Precip Type**: The type of precipitation recorded (e.g., "rain").

**Temperature** **(C):** The temperature in degrees Celsius.

**Apparent Temperature (C)**: The "feels-like" temperature, or how the temperature actually feels to a person, factoring in wind chill or humidity.

**Humidity**: The relative humidity percentage (values range from 0 to 1).

**Wind Speed (km/h):** The speed of the wind in kilometers per hour.

**Wind Bearing (degrees**): The direction the wind is coming from, represented in degrees (0-360).

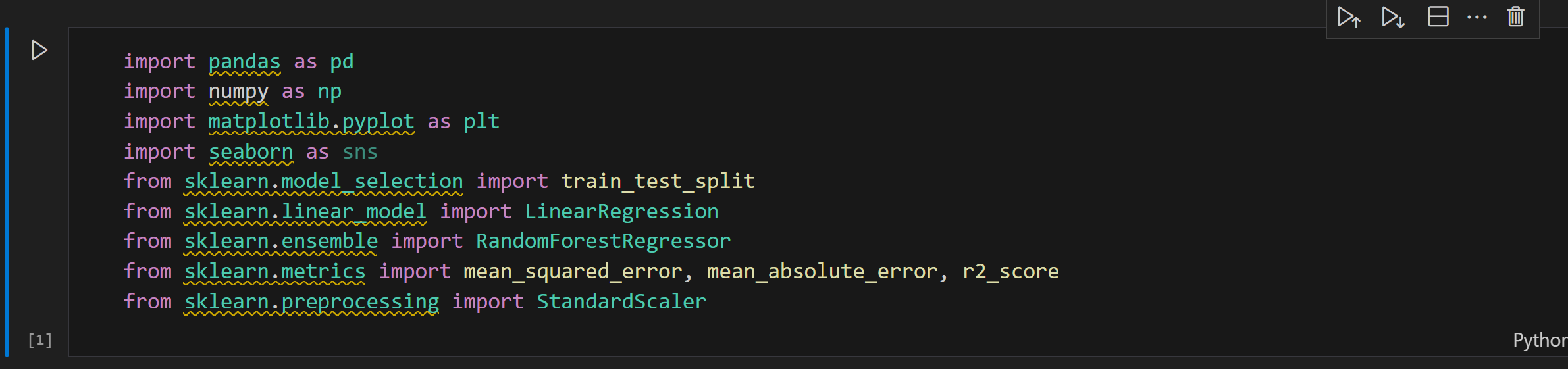
**Visibility (km)**: The visibility in kilometers, often influenced by factors like fog or heavy precipitation.

**Cloud Cover**: The percentage of the sky covered by clouds.

**Pressure (millibars):** The atmospheric pressure in millibars.

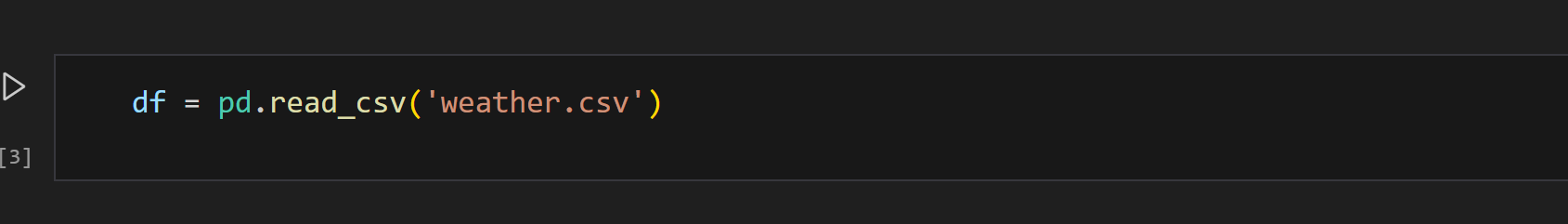
**Daily Summary**: A summary of the weather for the day.

**Importing all necessary Libraries**

****

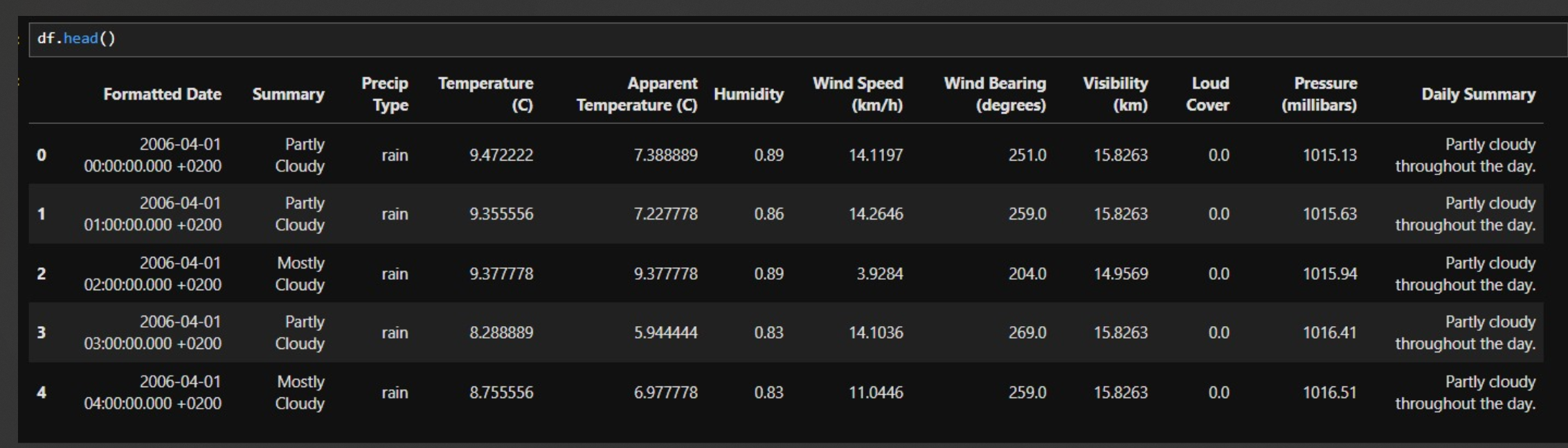
**Step 1: Load dataset**

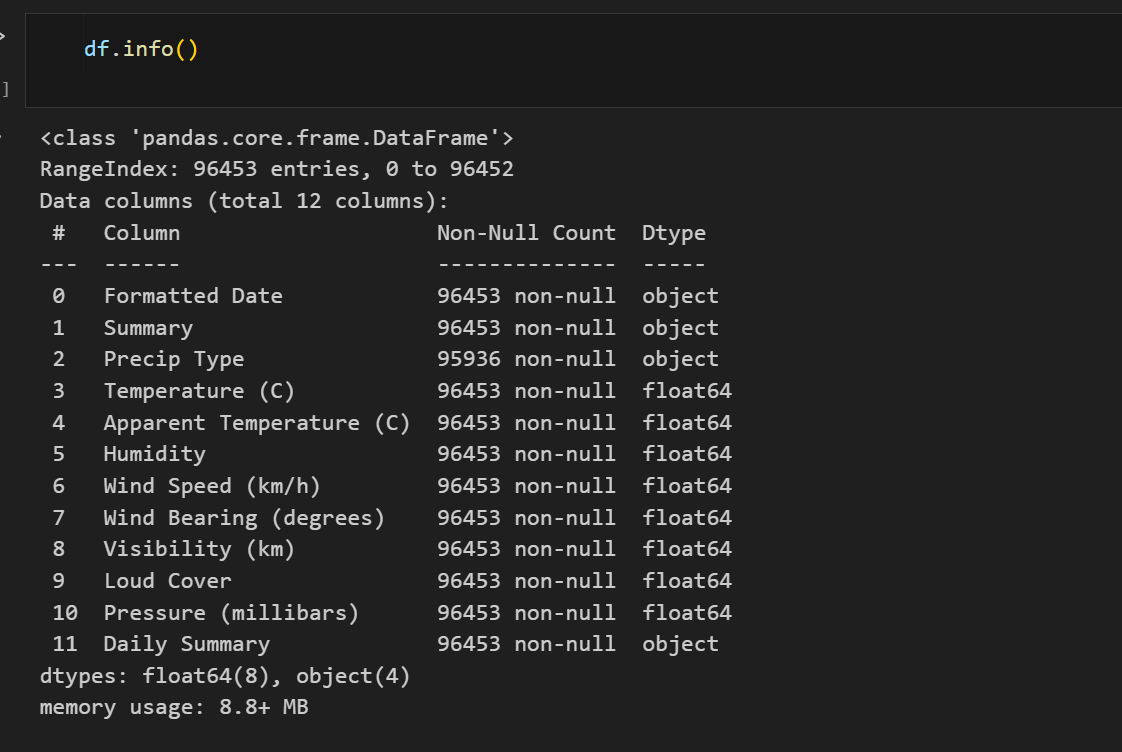
Here, we are loading our dataset weather.csv into a dataframe using **read\_csv()**

****

**Step 2: Exploring the dataset**

Here, we are glancing our dataset.

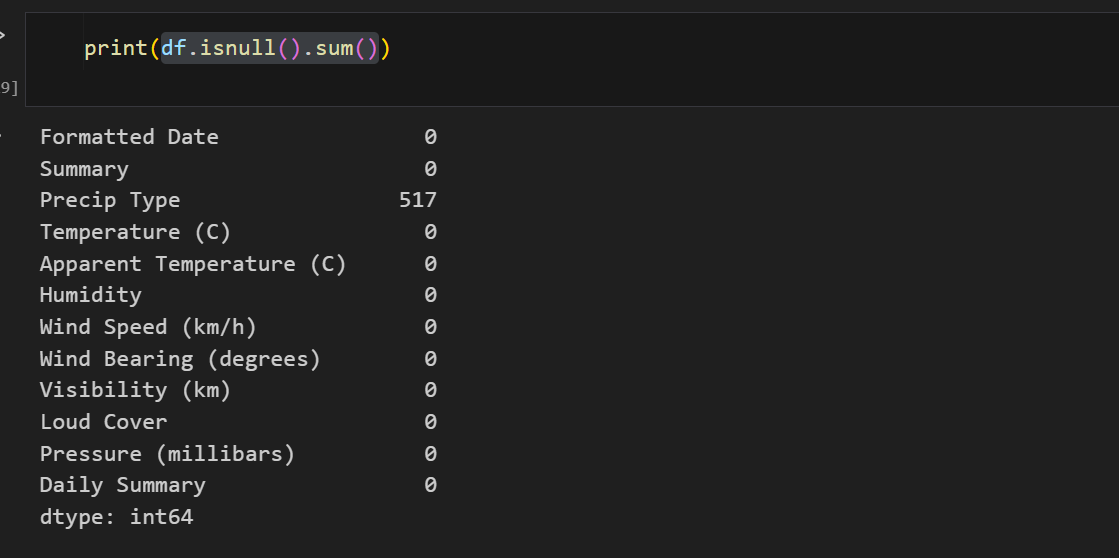




**Step 3: Data preprocessing**

**Checking for null values**

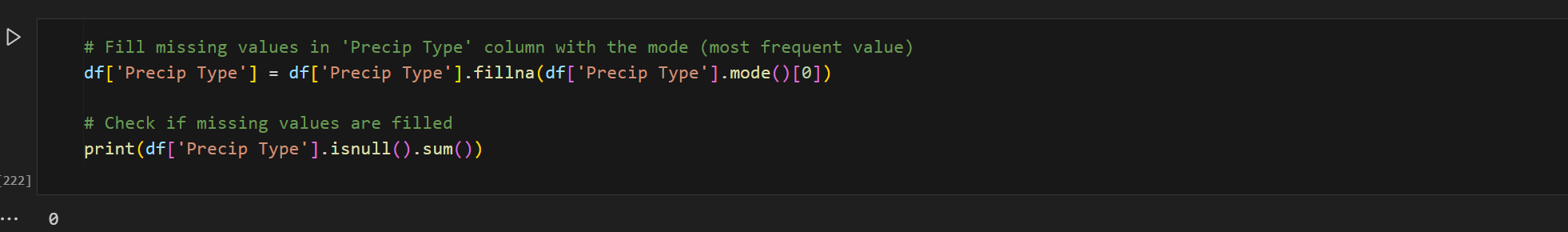
Here, we are checking sum of null values in each feature of dataframe using **df.isnull().sum()**



**Handle missing values (fill missing values with the column's mean )**

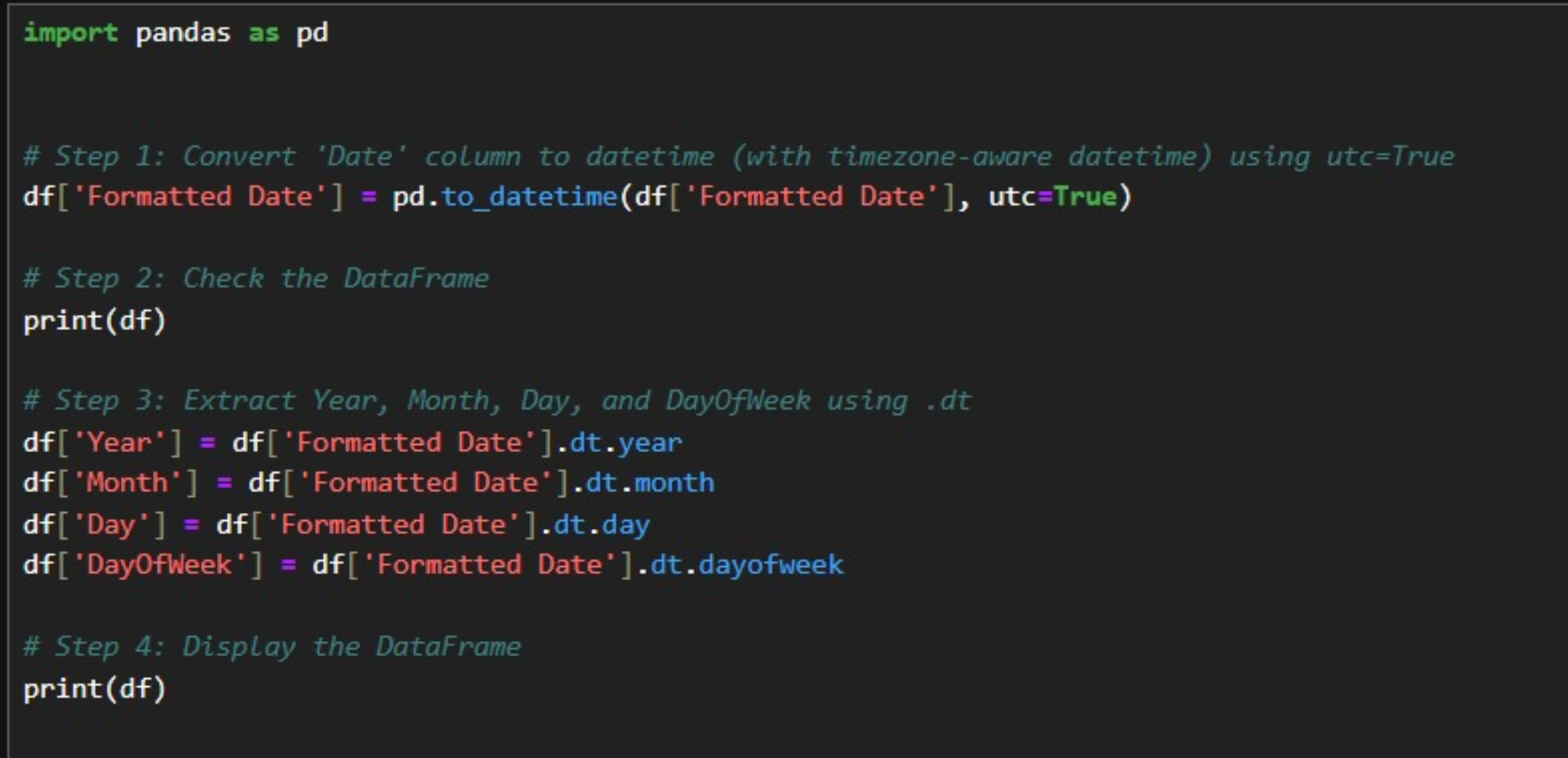
Here, feature Precip Type has 517 null values hence we populate these null values with most frequently occurring value.

**df['Precip Type'] = df['Precip Type'].fillna(df['Precip Type'].mode()[0])**

****

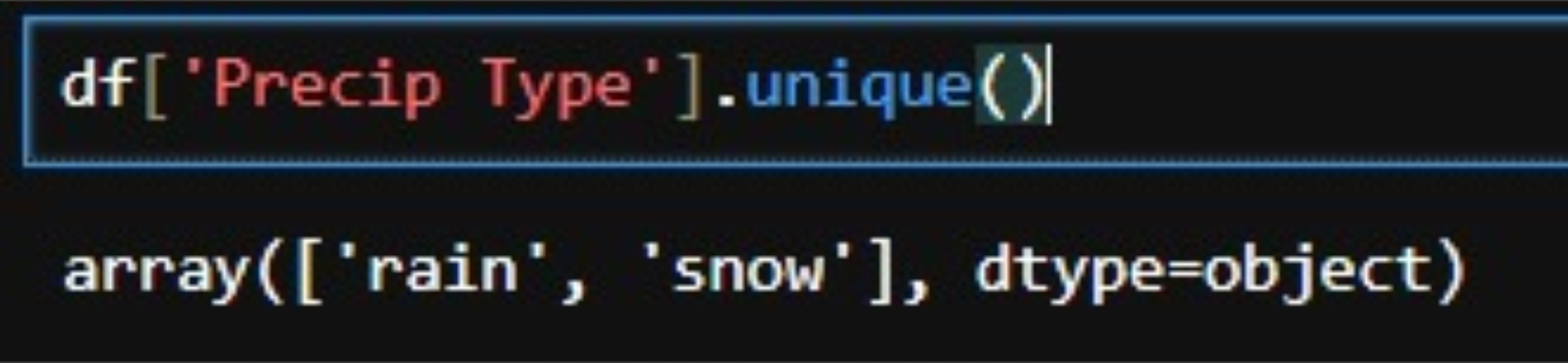
**Convert Date column to datetime and extract relevant features**

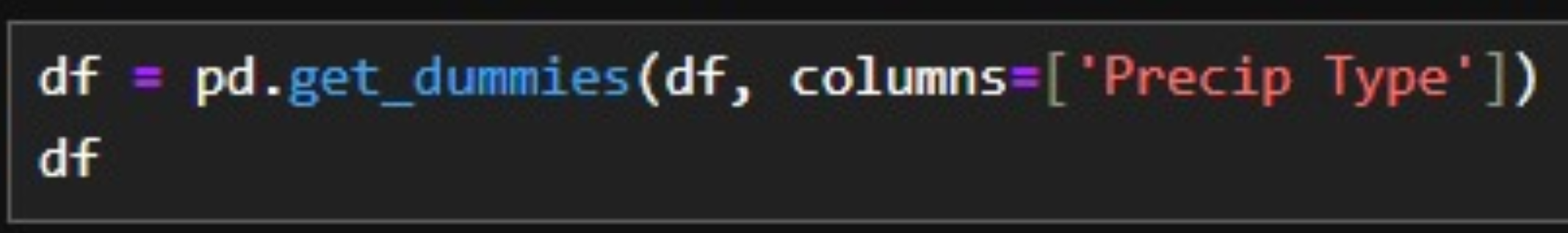
In this step we are extracting four new columns **( year, month, day, dayofweek)** from the formatted date feature.

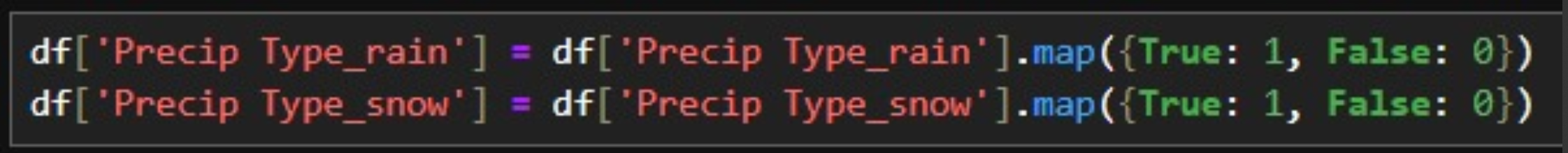


**Encoding the Precip Type column**

Here, we are using one hot encoding on feature Precip Type and then converting categoric data**(True, False)** to numeric data **(1,0)**

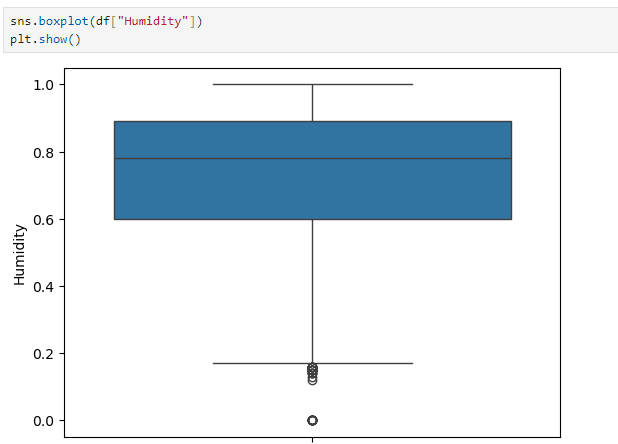
****

****

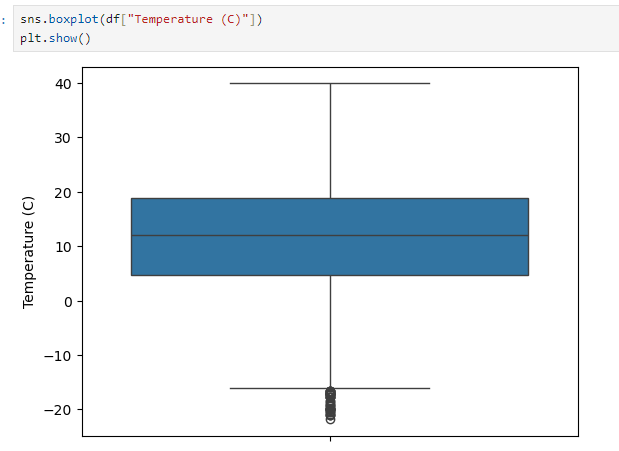
****

**Handling Outliers:**

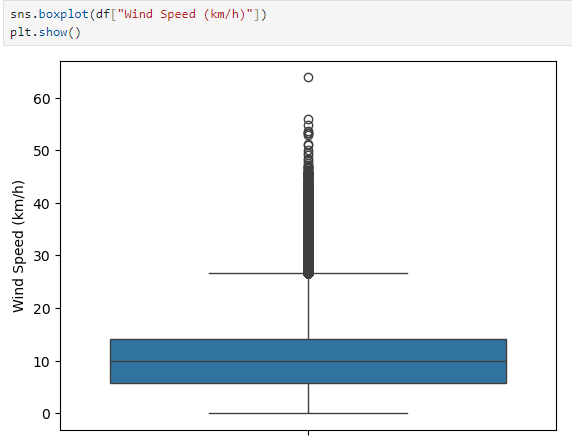
There are 4 Features in our dataset which have outliers as follows :

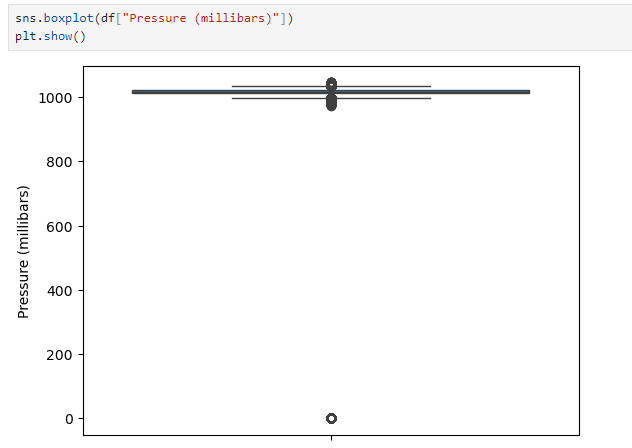
**Humidity**

**Temperature (C)**

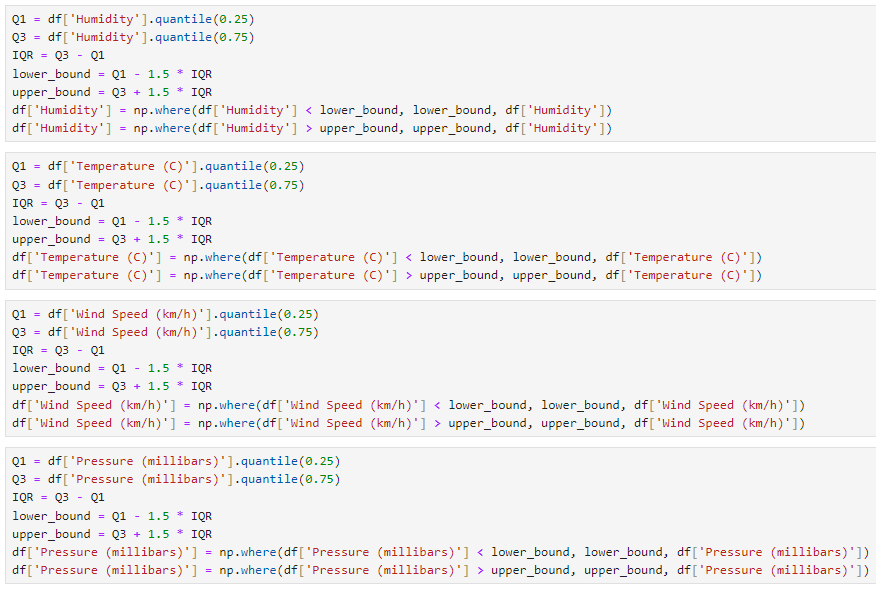
****

**Wind Speed (km/h)**

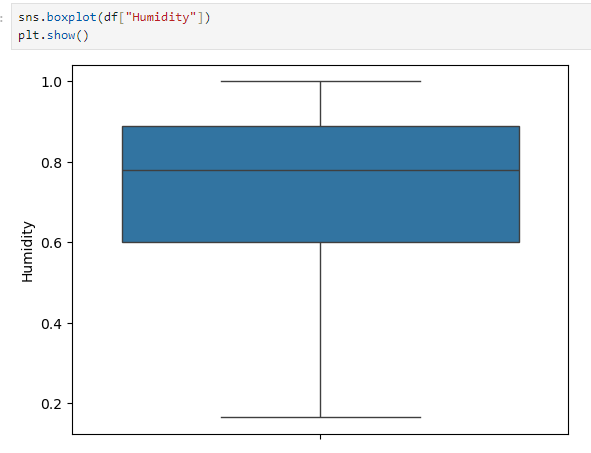
****

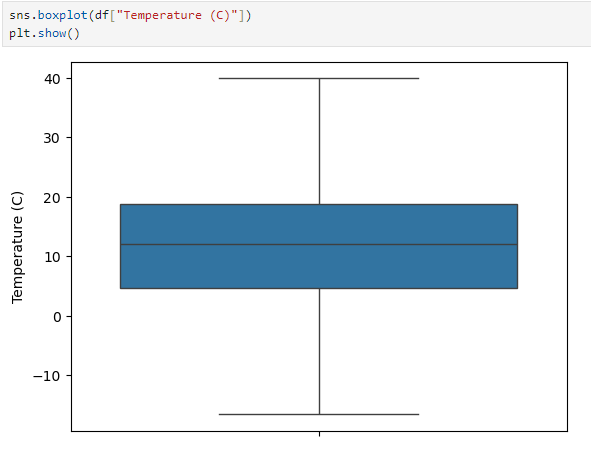
**Pressure (millibars)**

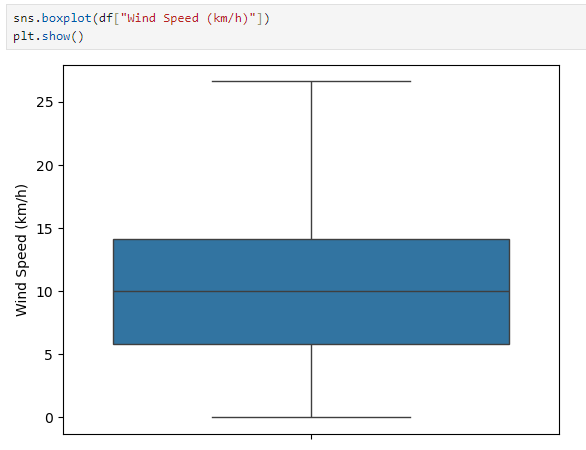
Here we have replaced our outliers with lower boundaries and Higher boundaries accordingly

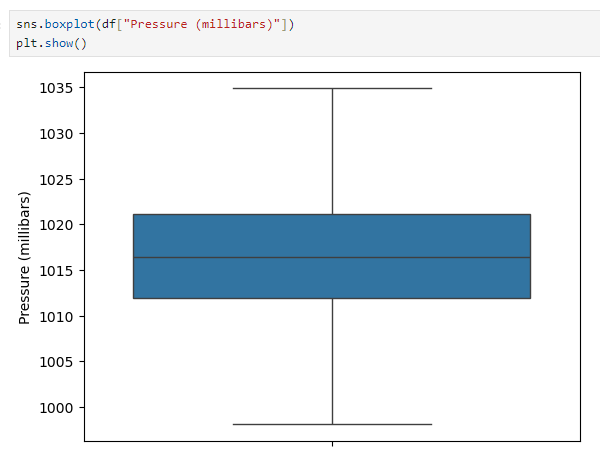


**Cross checking changes**

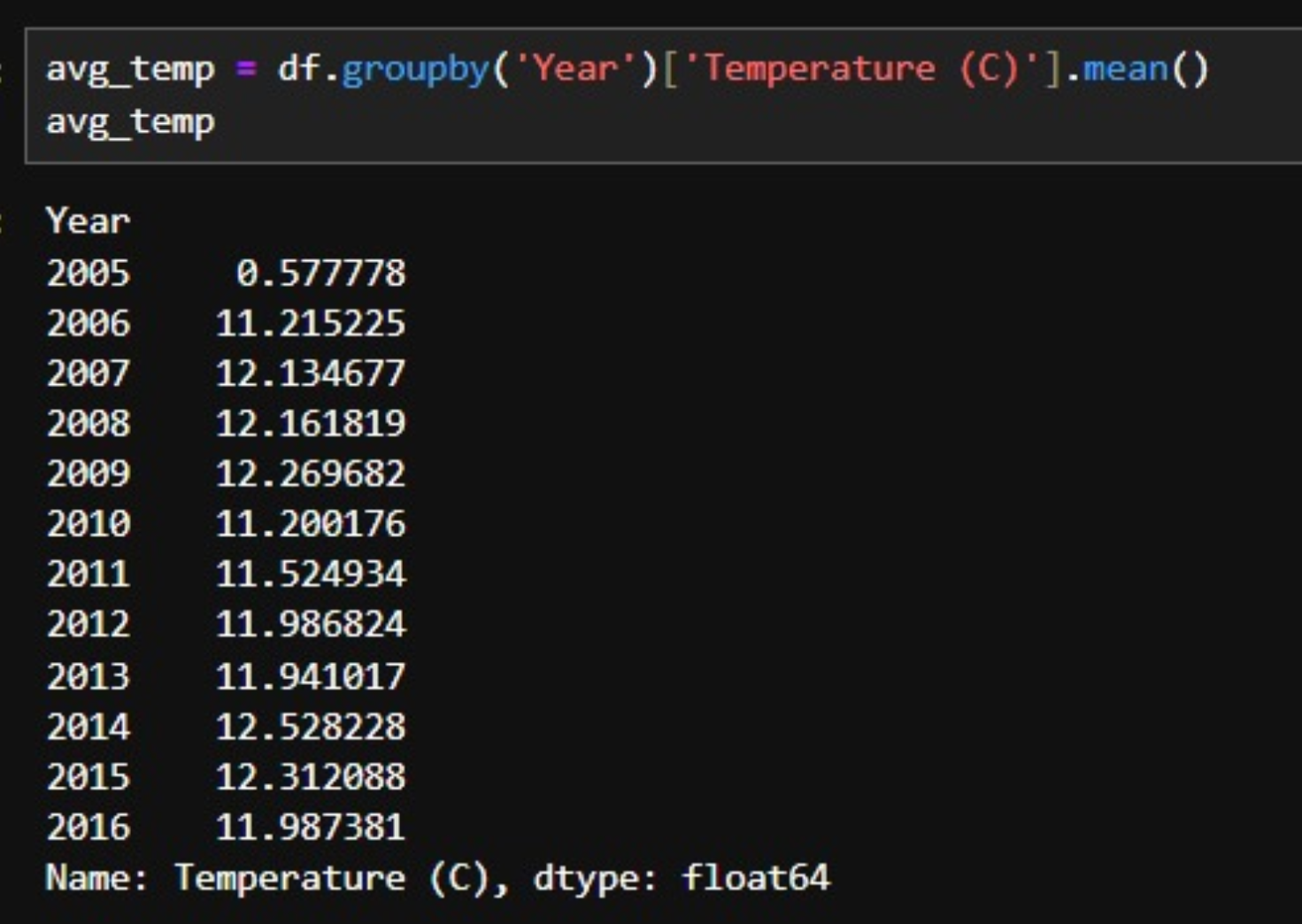
****

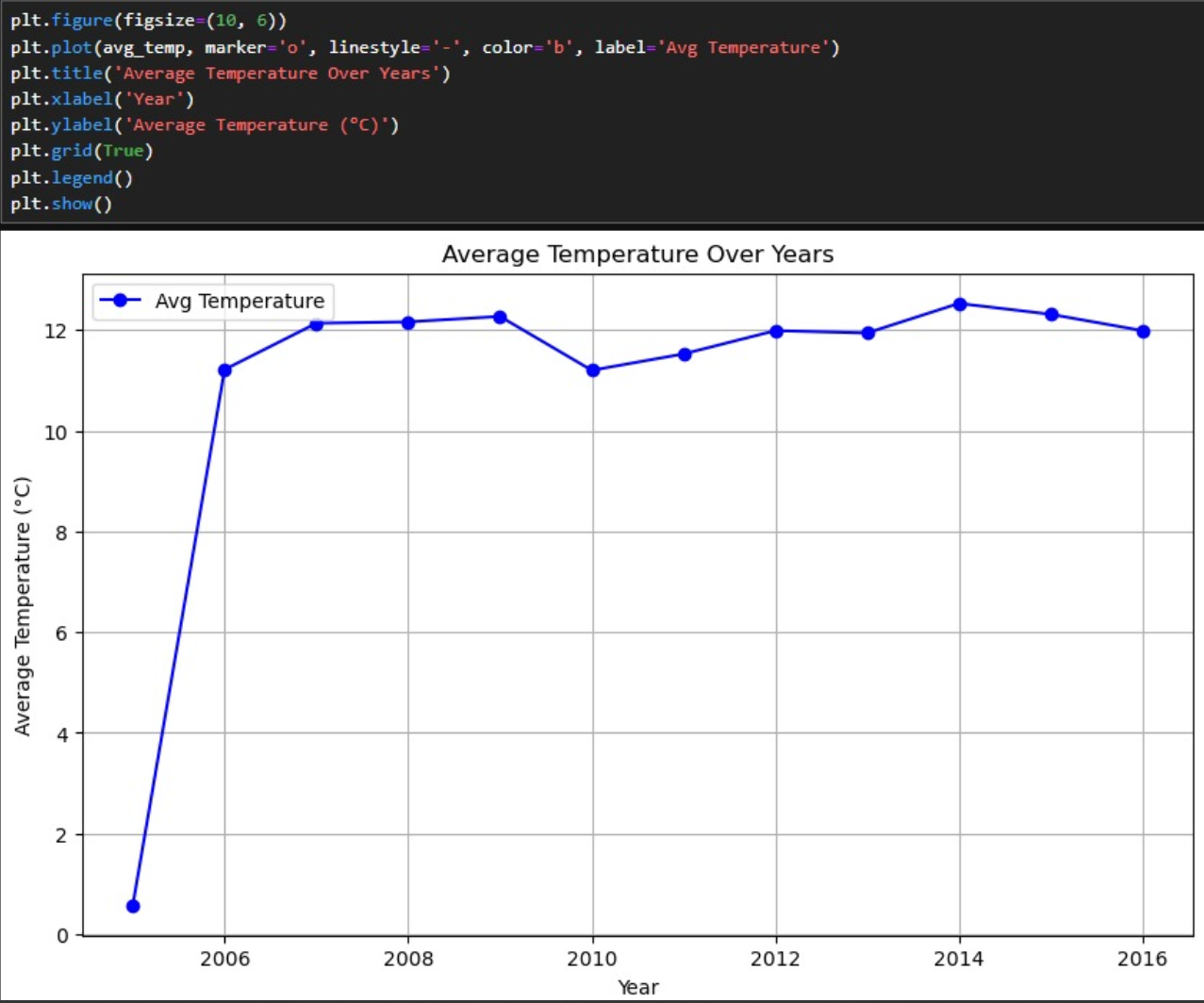
****

****

****

**Calculating Mean Temperature of each year and plot for average temperature and year**

****

****

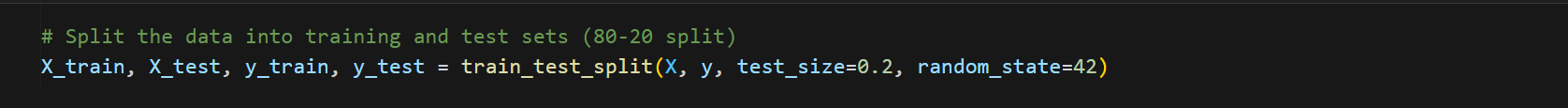
**Step 4 : Feature Selection**

Out of 17 features, we have taken 9 features and one target which is temperature.



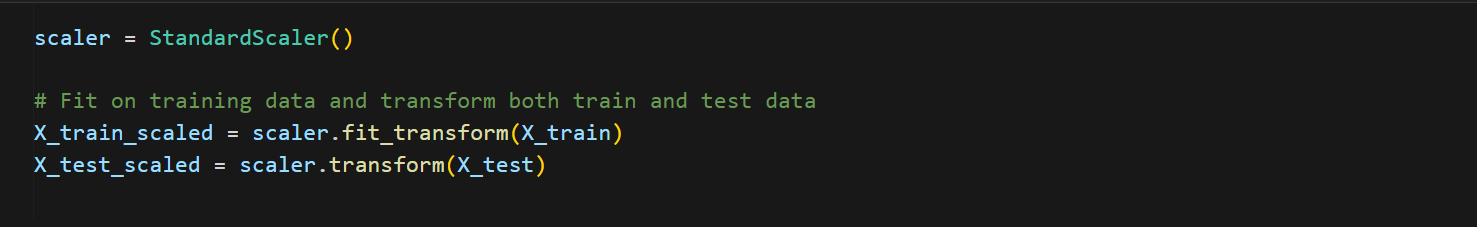
**Step 5: Split Data into Training and Test Sets**

Here, we are splitting our dataset into train and test with 80:20 ratios.



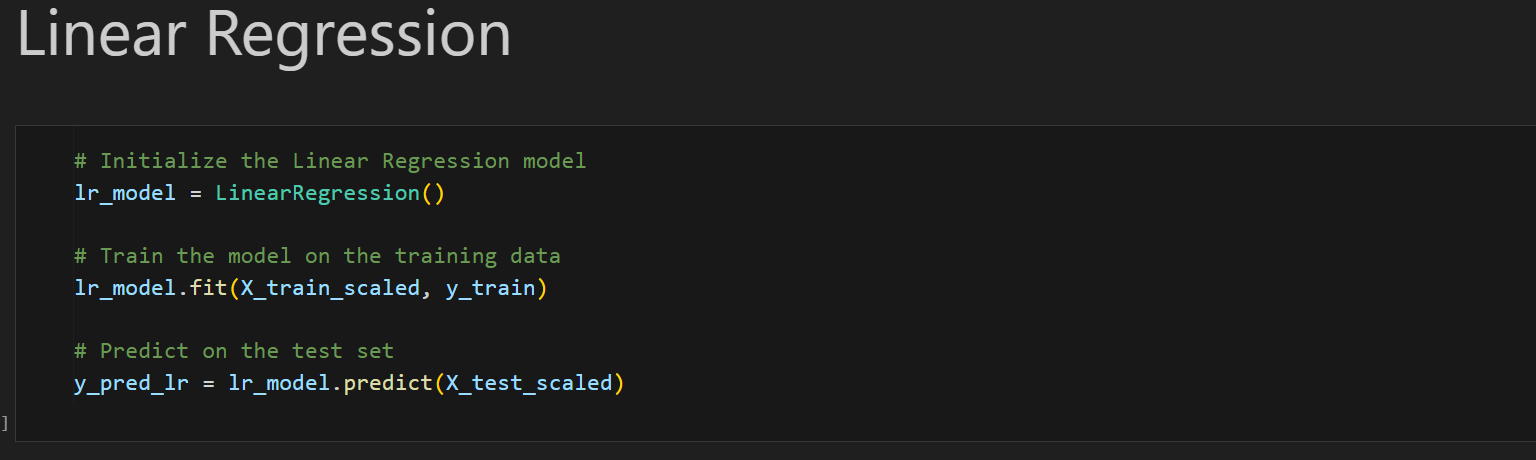
**Step 6: Scale the Data (Optional)**

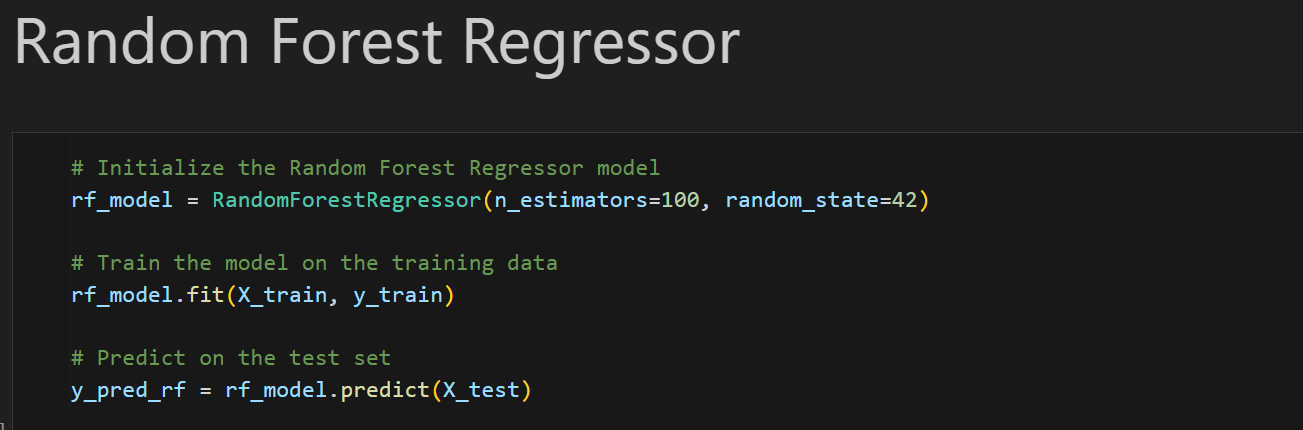
Scaling ensures that all features are on a comparable scale, which can improve the performance, speed, and stability of many machine learning algorithms. By standardizing or normalizing your data, you ensure that no feature dominates due to its range, making the learning process more efficient and effective.



**Step 7: Model Selection and Training**

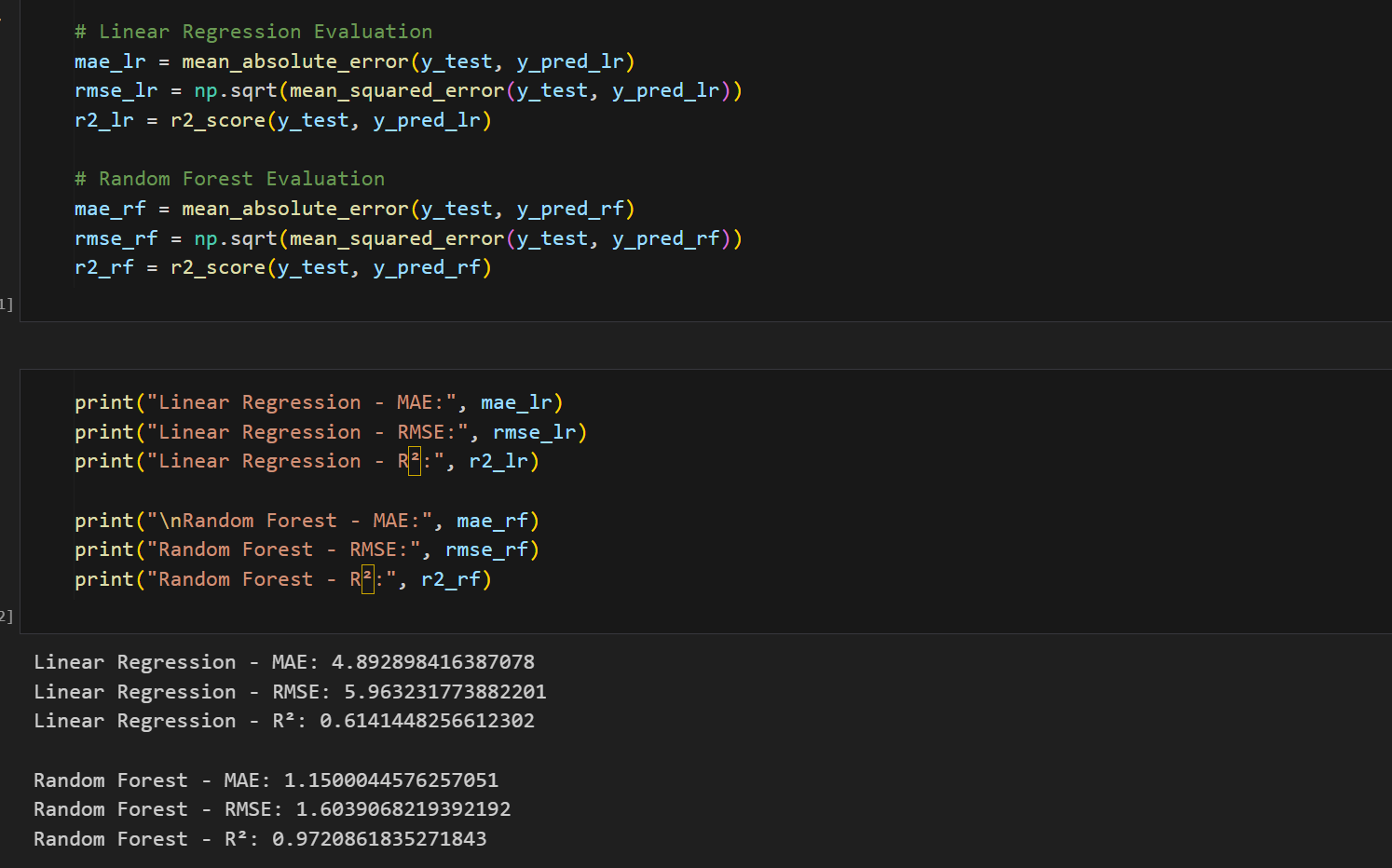
Since, our task is regression task we have chose two models which are Linear Regression and Random Forest Regressor.





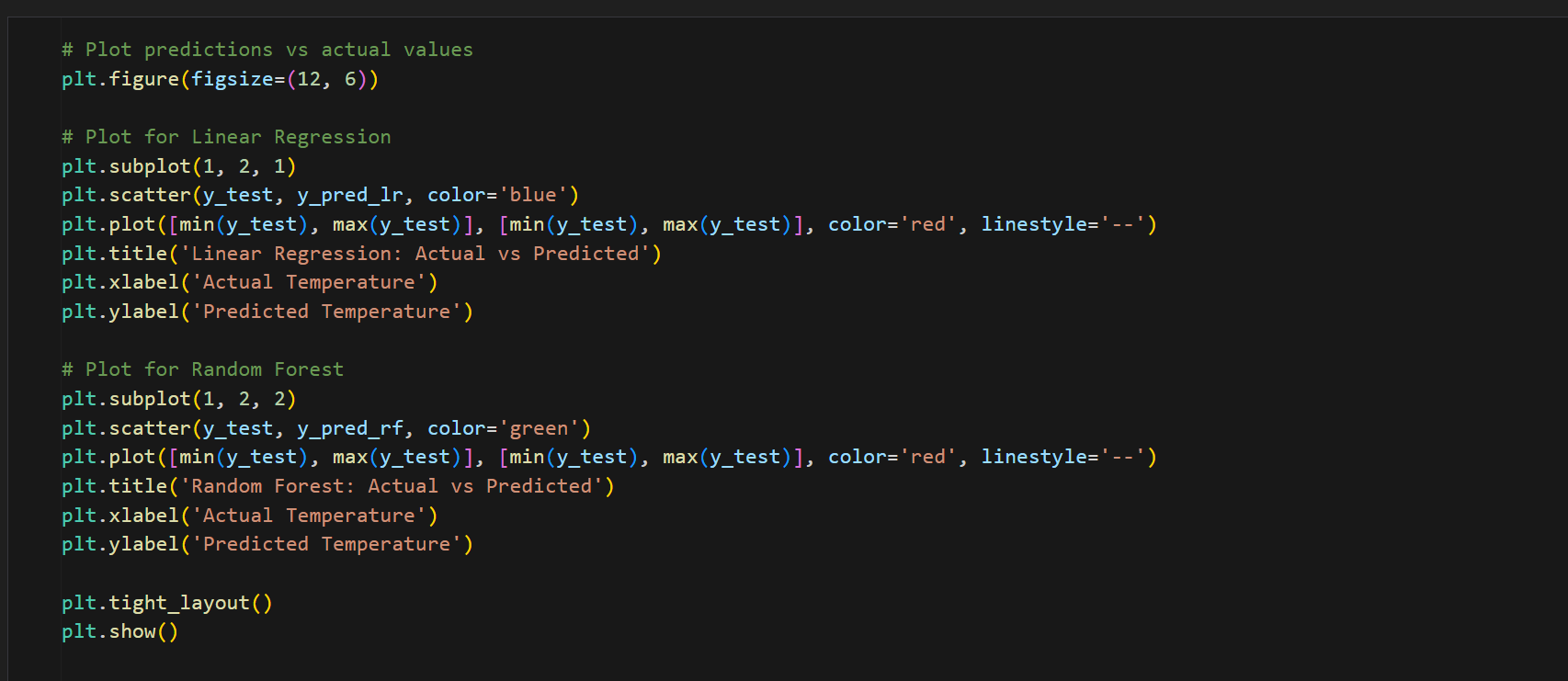
**Step 8: Evaluate the Model**

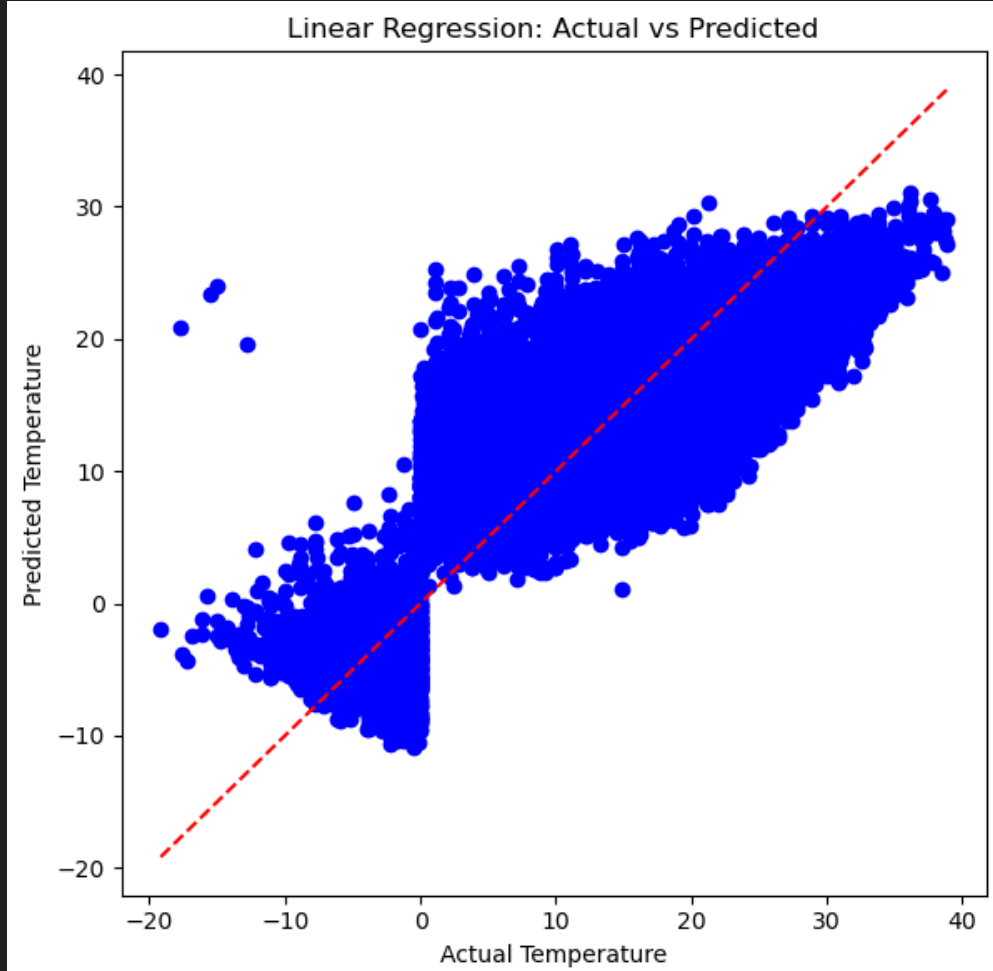
Once model is trained we have predicted outcomes for the test data set and found mean\_absolute\_errors, mean\_square\_error and R2\_score.



**Step 9: Visualize Predictions vs. Actual Values**

We are plotting graph for Actual values and Predicted values of temperature

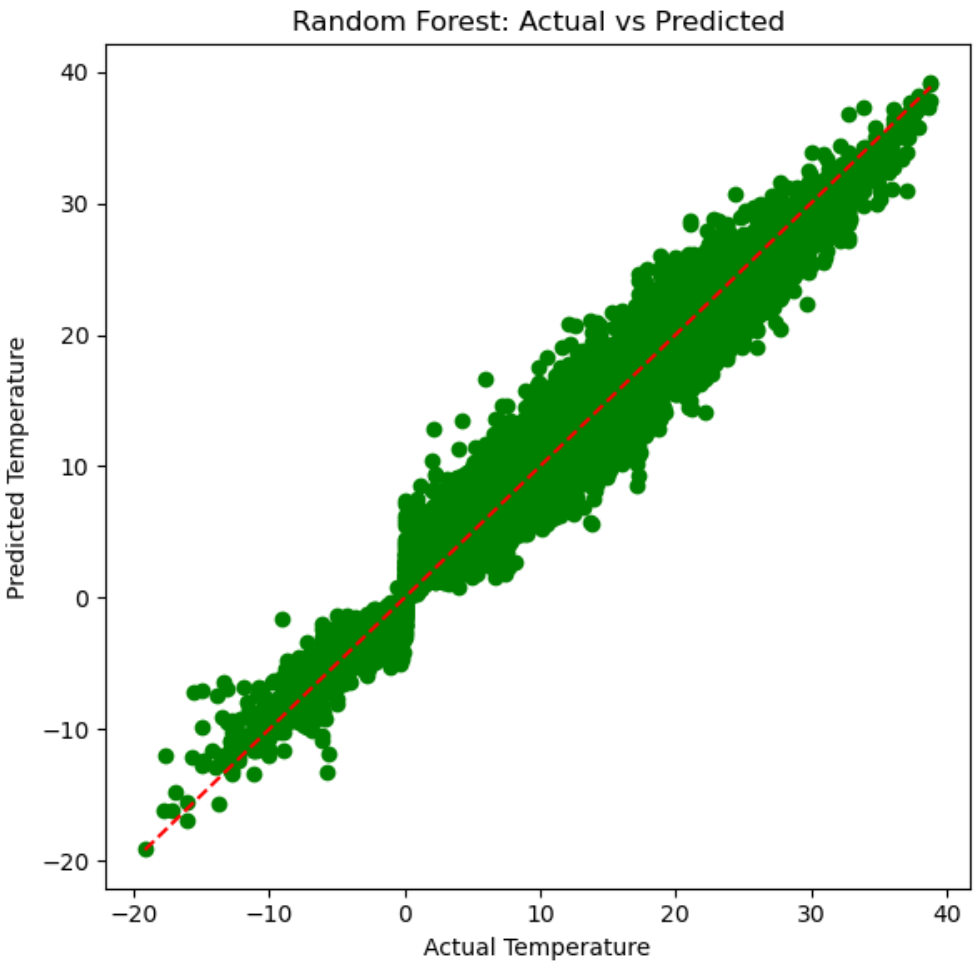




Since R2\_score of linear regression is 61% our model is really poor in prediction.

As you closely observe the plot above values are widely spread across the identity line which implies predicted value is way far from the actual observed value.

There is considerable deviation of predicted values from the actual values, hence our model linear regression performed poorly.



R2\_score for Random Forest Regressor is 97%

As you closely observe the graph the predicted values are converging towards the actual observed values and data is not widely spread across the identity line.

Having less deviation of predicted values from the actual values indicates our model Random Forest Regressor really well.

**Conclusion:**

The primary reason why Random Forest performed better than Linear Regression in your case is likely the non-linearity of the relationships in the data, as well as the ability of Random Forest to capture feature interactions and handle complex patterns without requiring explicit modeling. Linear regression, on the other hand, would have struggled with these aspects due to its assumption of linearity, sensitivity to multicollinearity, and potential underfitting in the presence of non-linear relationships.