EXPERIMENT 3  
  
**AIM** :

To develop a python program for linear regression model using time series dataset.  
  
  
**CODE** :

import pandas as pd

import numpy as np

from sklearn.model\_selection import train\_test\_split

from sklearn.linear\_model import LinearRegression

from sklearn.metrics import mean\_squared\_error

# Load the dataset

file\_path = '/mnt/data/MLTempDataset.csv'

data = pd.read\_csv(file\_path)

# Convert 'Datetime' to pandas datetime format

data['Datetime'] = pd.to\_datetime(data['Datetime'])

# Ensure data is sorted by time

data = data.sort\_values('Datetime')

# Feature engineering: converting datetime into numerical values for model input

data['Time\_Index'] = np.arange(len(data))

# Define features and target variable

X = data[['Time\_Index']]

y = data['DAYTON\_MW']

# Split data into training and test sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, y, test\_size=0.2, random\_state=42)

# Train a linear regression model

model = LinearRegression()

model.fit(X\_train, y\_train)

# Forecasting on the test set

y\_pred = model.predict(X\_test)

# Model evaluation

mse = mean\_squared\_error(y\_test, y\_pred)

print(f"Mean Squared Error of the Linear Regression Model: {mse}")

import matplotlib.pyplot as plt

import seaborn as sns

# Visualizing the model fit using a heatmap

plt.figure(figsize=(10, 5))

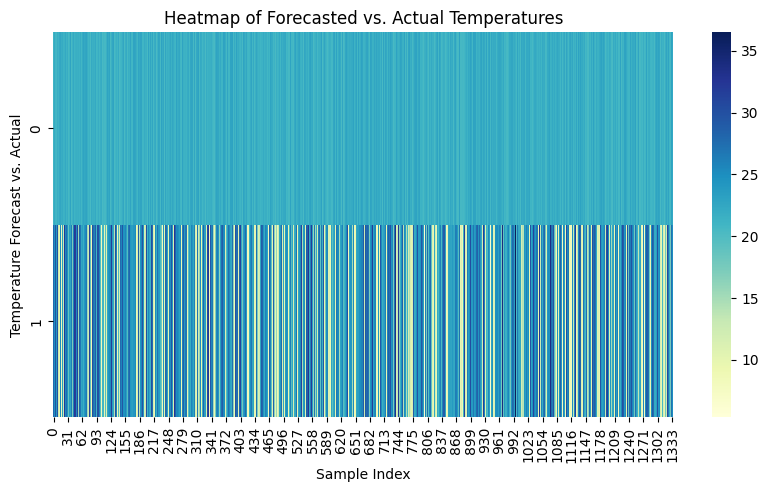
sns.heatmap([y\_pred, y\_test.values], cmap='YlGnBu', cbar=True, annot=False)

plt.title('Heatmap of Forecasted vs. Actual Temperatures')

plt.xlabel('Sample Index')

plt.ylabel('Temperature Forecast vs. Actual')

plt.show()

**OUTPUT** :   
  
  
Linear regression model output : MSE = 41.2175637730894  
  
Visualization :   
   
  
  
  
**RESULT** :   
 Thus the program has been executed successfully .