**Implement Linear Regression model using Time series Data**

**EX:No.3**

**DATE: /02/25**

# AIM: To develop a Linear Regression model using a Time Series Data

# CODE AND DESCRIPTION:

# import pandas as pd

# import matplotlib.pyplot as plt

# import seaborn as sns

# # Load the dataset

# file\_path = "/content/AirPassengers.csv"  # Replace with the actual path to your CSV file

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

# # Display the first few rows of the dataset

# print(data.head())

# # Convert the 'Month' column to datetime format

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

# # Set the 'Month' column as the index

# data.set\_index('Month', inplace=True)

# # Plotting the time series data

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

# plt.plot(data.index, data['#Passengers'], color='blue', marker='o', linestyle='-')

# plt.title('Air Passengers Over Time (1949-1960)')

# plt.xlabel('Month')

# plt.ylabel('Number of Passengers')

# plt.grid(True)

# plt.show()

# from statsmodels.tsa.seasonal import seasonal\_decompose

# # Perform seasonal decomposition

# decomposition = seasonal\_decompose(data['#Passengers'], model='multiplicative')

# # Plot the decomposition

# plt.figure(figsize=(12, 8))

# plt.subplot(4, 1, 1)

# plt.plot(decomposition.trend, color='blue')

# plt.title('Trend Component')

# plt.subplot(4, 1, 2)

# plt.plot(decomposition.seasonal, color='green')

# plt.title('Seasonal Component')

# plt.subplot(4, 1, 3)

# plt.plot(decomposition.resid, color='red')

# plt.title('Residual Component')

# plt.subplot(4, 1, 4)

# plt.plot(data['#Passengers'], color='purple')

# plt.title('Original Time Series')

# plt.tight\_layout()

# plt.show()

# 

# from sklearn.linear\_model import LinearRegression

# import numpy as np

# # Prepare the data for linear regression

# # Convert the datetime index to numerical values (e.g., number of months since the start)

# data['Month\_num'] = np.arange(len(data))

# # Define the features (X) and target (y)

# X = data[['Month\_num']]  # Feature: month number

# y = data['#Passengers']  # Target: number of passengers

# # Fit the linear regression model

# model = LinearRegression()

# model.fit(X, y)

# # Predict the values using the model

# y\_pred = model.predict(X)

# # Plot the original data and the linear regression line

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

# plt.plot(data.index, data['#Passengers'], color='blue', marker='o', linestyle='-', label='Original Data')

# plt.plot(data.index, y\_pred, color='red', linestyle='--', label='Linear Regression')

# plt.title('Air Passengers Over Time (1949-1960) with Linear Regression')

# plt.xlabel('Month')

# plt.ylabel('Number of Passengers')

# plt.legend()

# plt.grid(True)

# plt.show()

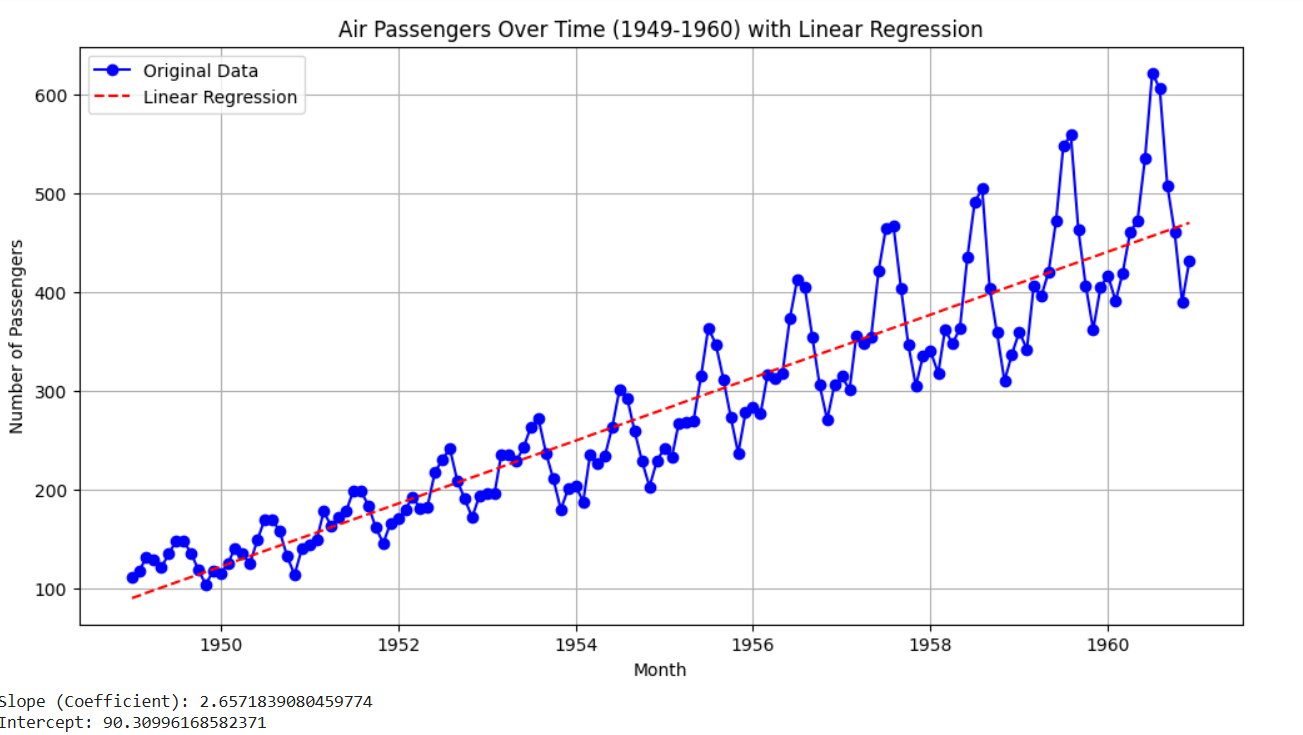
# # Display the slope and intercept of the linear regression model

# print(f"Slope (Coefficient): {model.coef\_[0]}")

# print(f"Intercept: {model.intercept\_}")

# 

# OUTPUT



**RESULT:**

Thus, the program has been completed and verified successfully.