DATE:23-01-25

EXPERIMENT - 1

IMPLEMENTING A PROGRAM FOR TIME SERIES DATA CLEANING , LOADING , HANDLING AND PREPROCESSING

AIM : To implement a program for time series data cleaning, loading and handling time series data and pre-processing techniques.

ALGORITHM :

Load Libraries
Load Dataset
Data Cleaning
Data Preprocessing
Data Visualization
ARIMA Model Training
Prediction
Visualization of Predictions
Model Accuracy Evaluation

CODE :

```python
import pandas as pd

# Assuming your dataset is named 'airline_passengers.csv'
df = pd.read_csv('/content/airline-passengers.csv', index_col='Month', parse_dates=True)

# Check for missing values
print(df.isnull().sum())

# Handle missing values (example: fill with the previous value)
df.fillna(method='ffill', inplace=True)

# Check for duplicates
print(df.duplicated().sum())

# Remove duplicates (if any)
df.drop_duplicates(inplace=True)

import matplotlib.pyplot as plt

plt.figure(figsize=(12, 6))
plt.plot(df.index, df['Passengers'])
plt.title('Airline Passengers Time Series')
plt.xlabel('Month')
plt.ylabel('Passengers')
plt.grid(True)
plt.show()

!pip install statsmodels==0.13.5 # Install the statsmodels library

import statsmodels.api as sm
```

```python
from statsmodels.tsa.arima.model import ARIMA

# (p, d, q) are the model orders
# Replace with the appropriate values for your data
model = ARIMA(df['Passengers'], order=(5, 1, 0))

model_fit = model.fit()

# Define the number of steps (periods) to forecast ahead
steps = 12  # For example, predict the next 12 months

# Now you can make the predictions:
predictions = model_fit.predict(start=len(df), end=len(df) + steps - 1)

import matplotlib.pyplot as plt

# Plot the original time series data
plt.figure(figsize=(12, 6))
plt.plot(df.index, df['Passengers'], label='Actual')

# Plot the predicted values
plt.plot(pd.date_range(start=df.index[-1], periods=steps + 1, freq='MS')[1:], predictions,
label='Predictions', color='red')
# 'MS' frequency stands for Month Start

plt.title('Airline Passengers Time Series with Predictions')
plt.xlabel('Month')
plt.ylabel('Passengers')
plt.legend()
plt.grid(True)
plt.show()
```
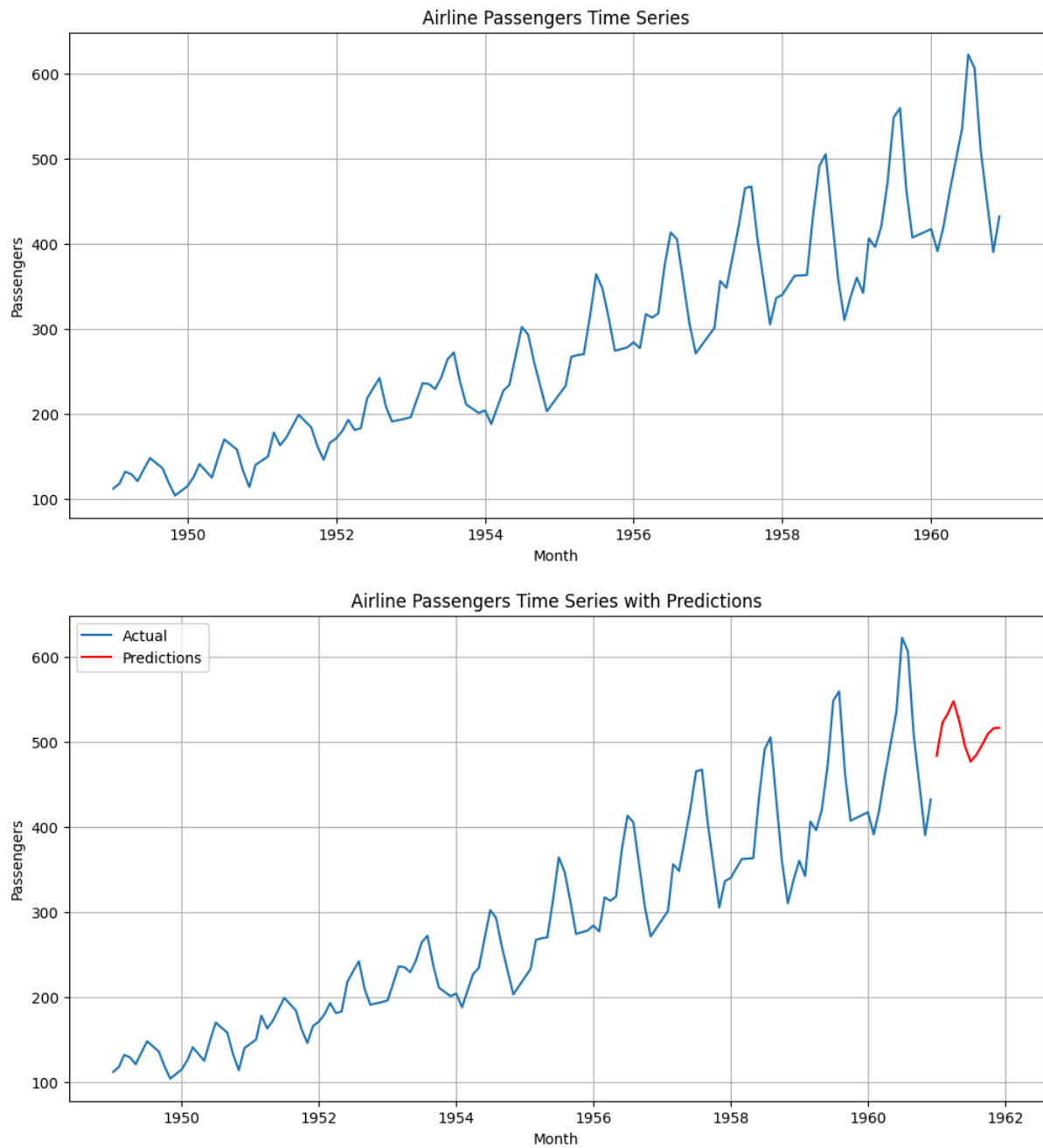
Airline Passengers Time Series

Airline Passengers Time Series with Predictions

RESULT : Thus the program has been successfully implemented.