

EXERCISE NO. 2

VISUALISATION OF TIME SERIES DATA

AIM:

To implement programs for visualising time series data.

ALGORITHM:

1. Import the necessary libraries.
2. Load the dataset.
3. Implement visualisation charts such as histogram, line chart, box, plot, heatmap, autocorrelation plot, partial autocorrelation plot and seasonal decomposition plot.

PROGRAM:

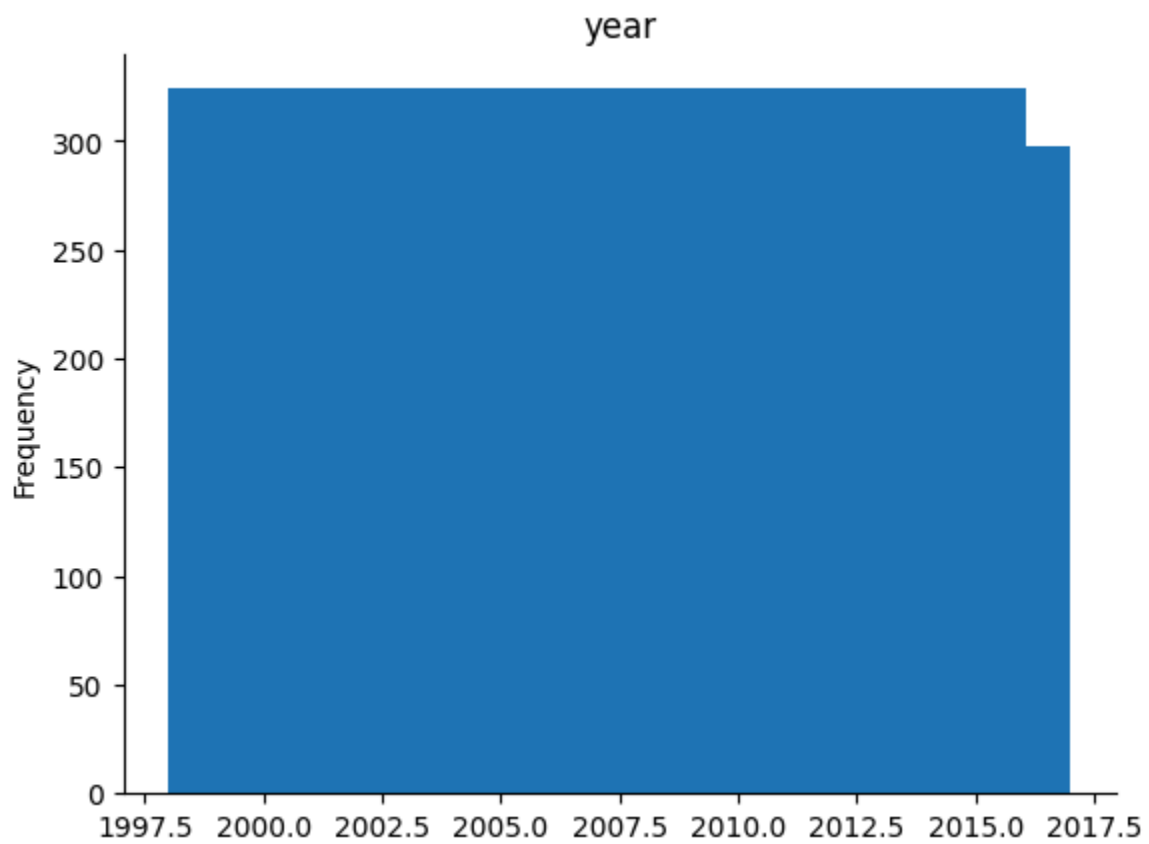
```
import pandas as pd
from pandas.plotting import autocorrelation_plot
from pandas.plotting import lag_plot
import matplotlib.pyplot as plt
from statsmodels.tsa.seasonal import seasonal_decompose

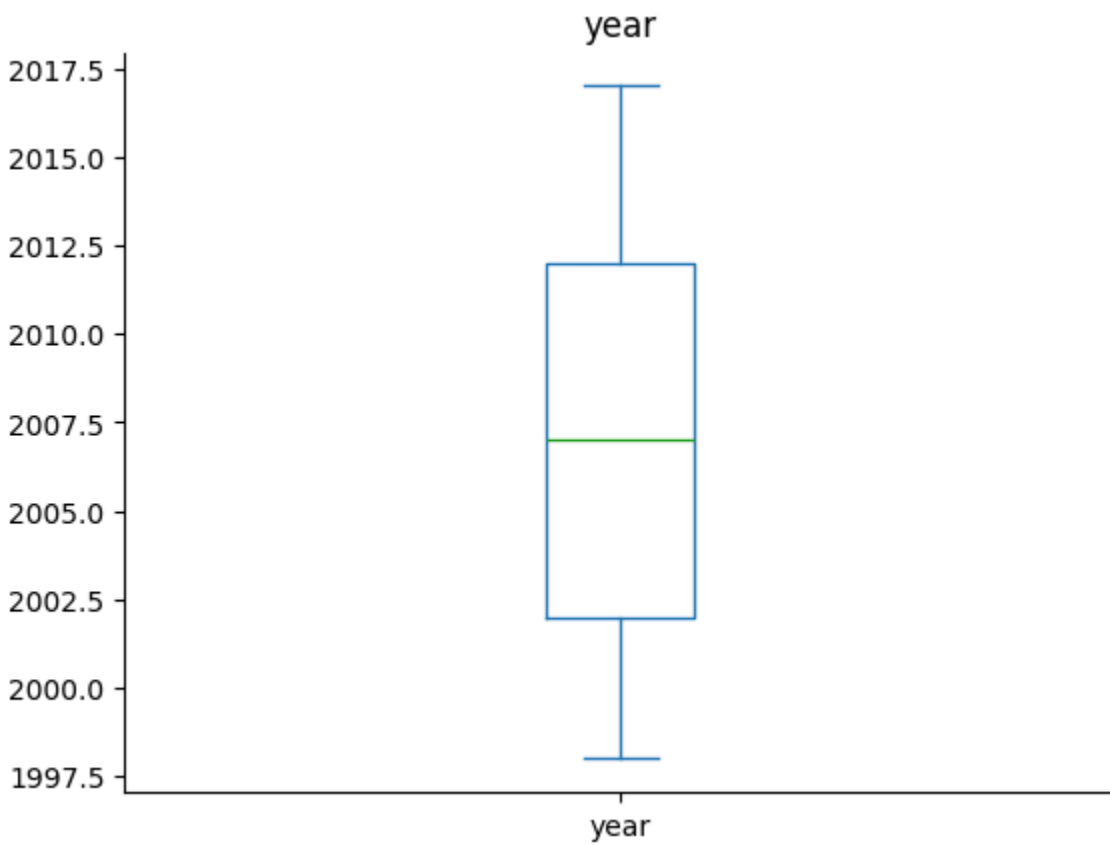
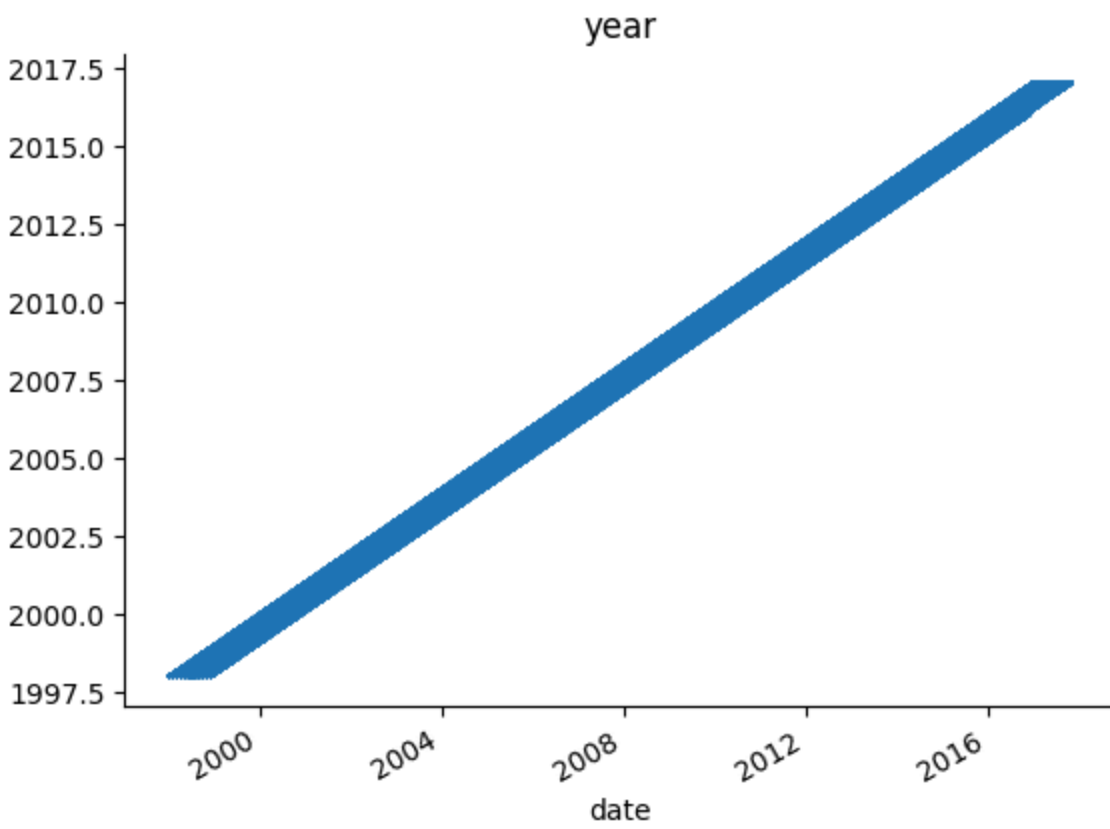
df = pd.read_csv('/amazon.csv', encoding='latin1')
month_map = {
    'Janeiro': 'January', 'Fevereiro': 'February', 'Março': 'March',
    'Abril': 'April', 'Maio': 'May', 'Junho': 'June',
    'Julho': 'July', 'Agosto': 'August', 'Setembro': 'September',
    'Outubro': 'October', 'Novembro': 'November', 'Dezembro': 'December'
}

df['month'] = df['month'].map(month_map)
df['date'] = pd.to_datetime(df['month'] + ' ' + df['year'].astype(str),
format='%B %Y')
df.set_index('date', inplace=True)
df['year'].plot(kind='hist', bins=20, title='year')
plt.gca().spines[['top', 'right']].set_visible(False)
df['year'].plot(kind='line', title='year')
plt.gca().spines[['top', 'right']].set_visible(False)
df['year'].plot(kind='box', title='year')
plt.gca().spines[['top', 'right']].set_visible(False)
numerical_df = df.select_dtypes(include=['number'])
numerical_df.corr().style.background_gradient(cmap='coolwarm')
autocorrelation_plot(df['year'])
```

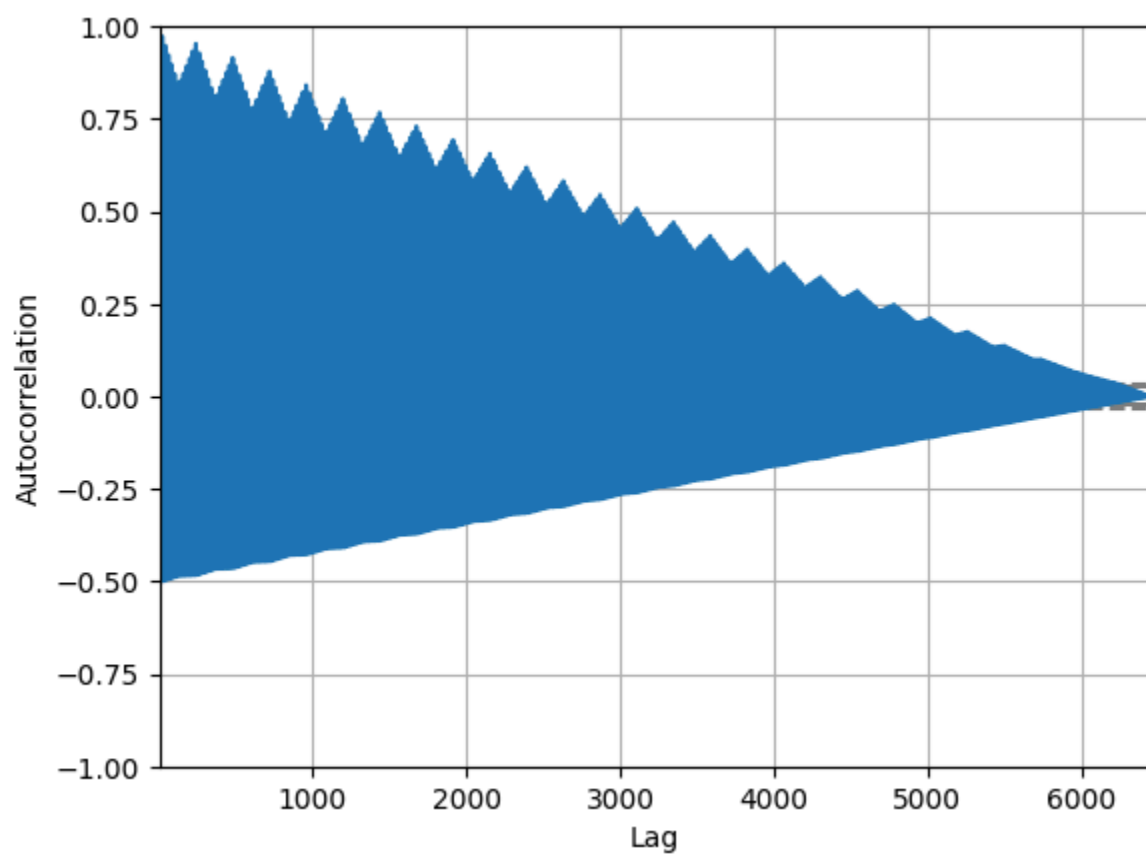
```
plt.show()
lag_plot(df['year'])
plt.show()
result = seasonal_decompose(df['year'], model='additive', period=12)
result.plot()
plt.show()
```

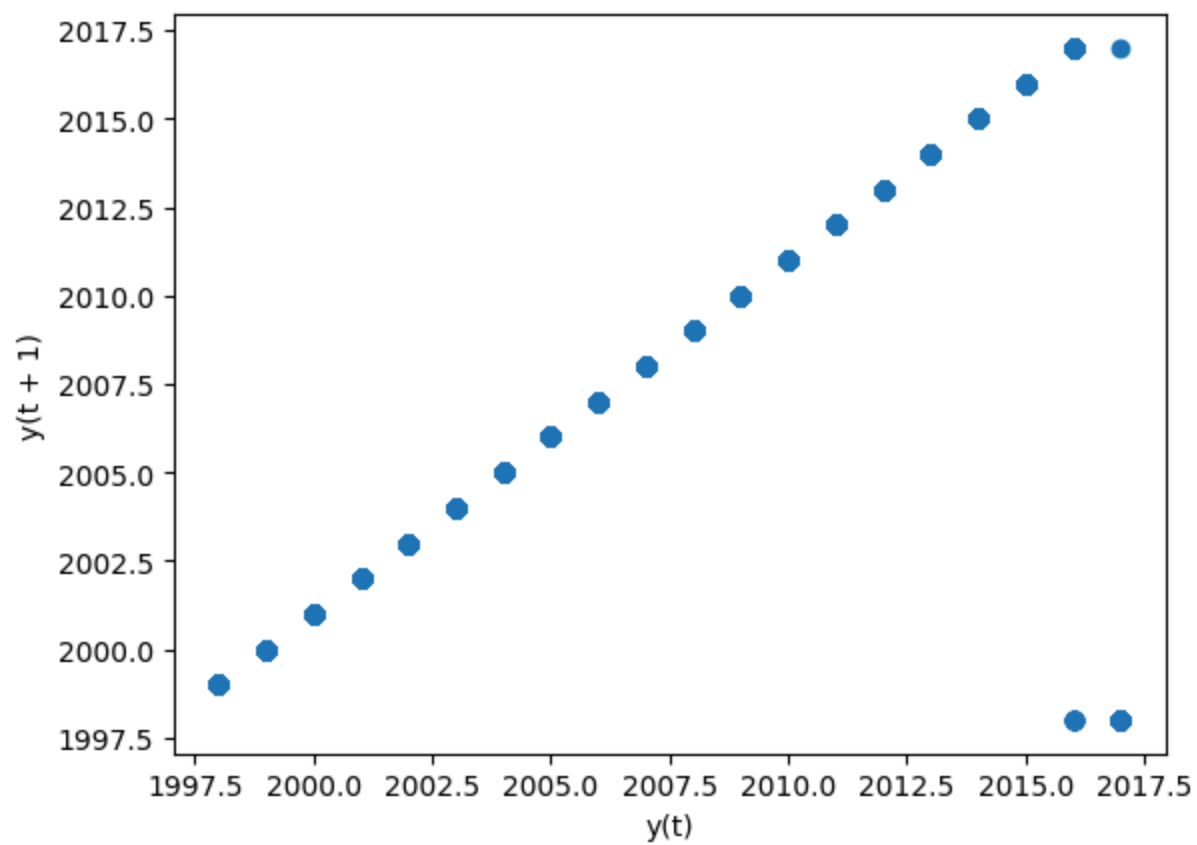
OUTPUT:

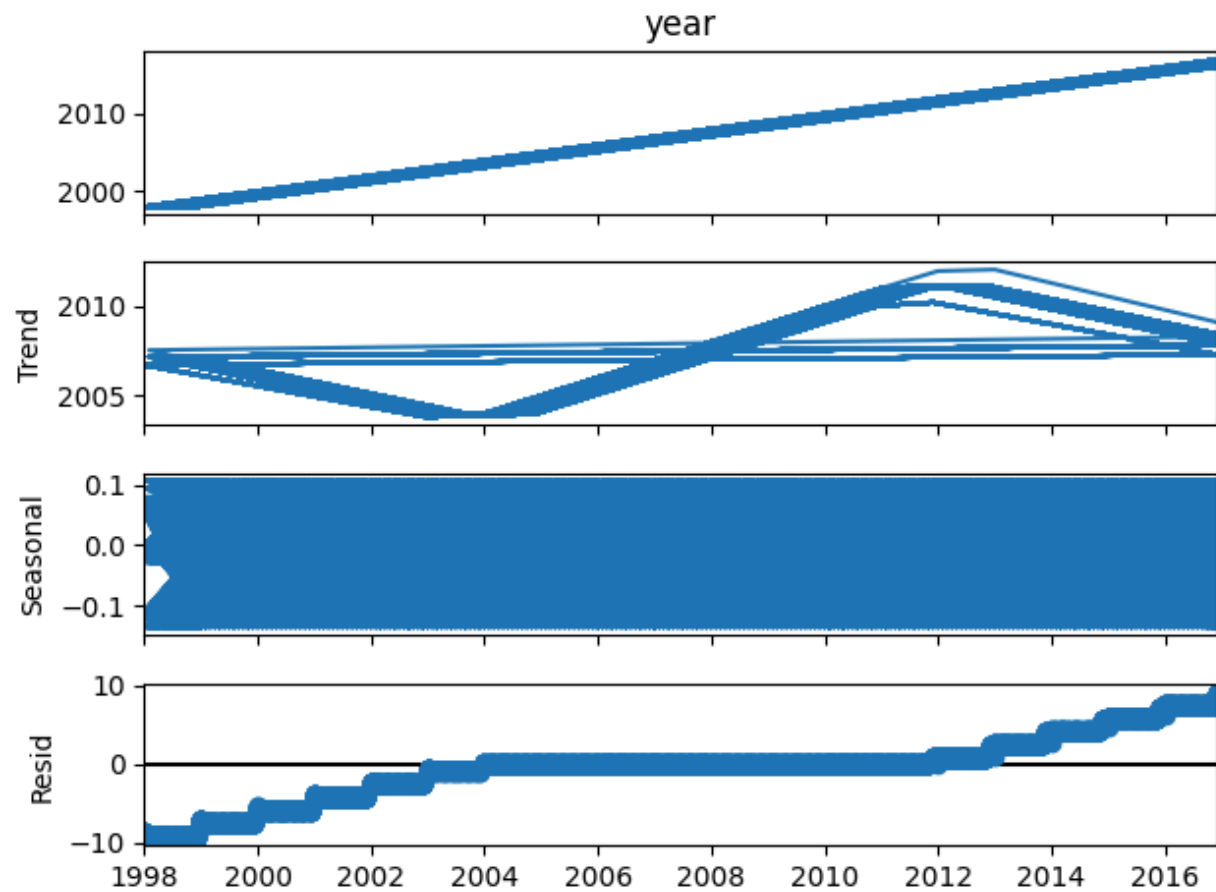




	year	number
year	1.000000	0.065201
number	0.065201	1.000000







RESULT:

Thus the program has been successfully implemented and verified.