**Name: Abhay Sharma**

**Class: BE - C**

**Moodle Id: 20102065**

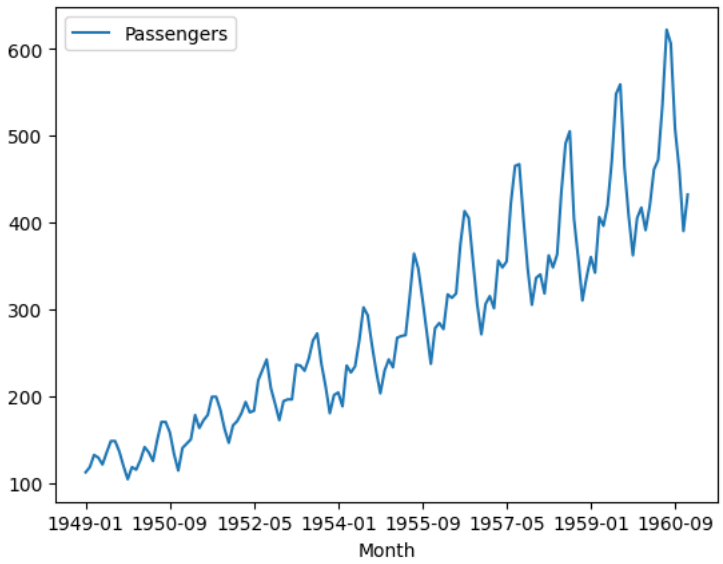
**Roll No.: 28**

**Sub.: Applied Data Science**

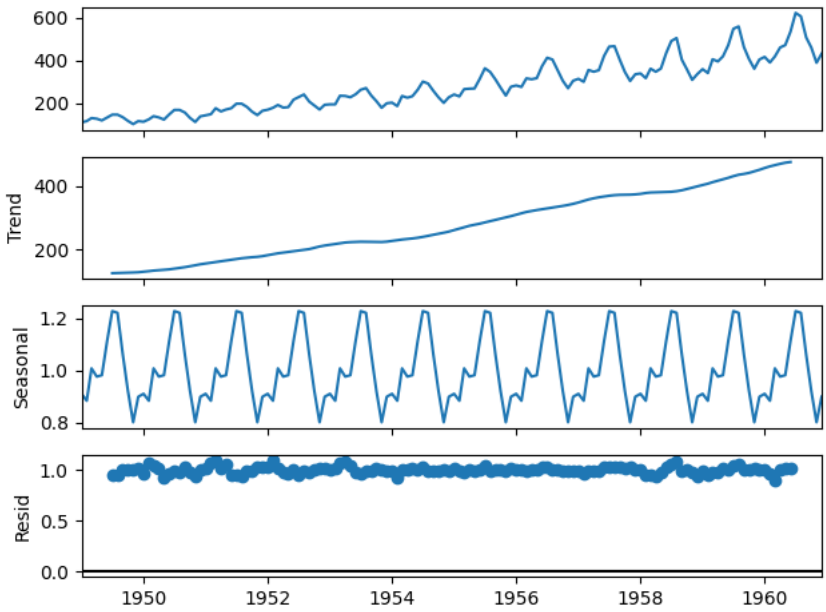
# Experiment No.: 09

```
import pandas as pd
import matplotlib.pyplot as plt
%matplotlib inline
```

```
from matplotlib import pyplot
series=pd.read_csv(r'/content/airlinepassenger.csv', header=0, index_col=0)
series.plot()
pyplot.show()
```




```
from matplotlib import pyplot
from statsmodels.tsa.seasonal import seasonal_decompose
data = pd.read_csv(r'/content/airlinepassenger.csv')
data.columns= ['date', 'data']
#Change datatype to pandas datetime
data['date'] = pd.to_datetime (data['date'])
data=data.set_index('date')
result = seasonal_decompose(data, model='multiplicative')
result.plot()
pyplot.show()
```




```
import pandas as pd
import numpy as np

product= {'month': [1,2,3,4,5,6,7,8,9,10,11,12], 'demand': [290,260,288,300,310,303,329,340,316,330,308 ,310]}

df = pd.DataFrame (product)

df.head()
```

| | month | demand |
|---|---|---|
| **0** | 1 | 290 |
| **1** | 2 | 260 |
| **2** | 3 | 288 |
| **3** | 4 | 300 |
| **4** | 5 | 310 |

Next steps: View recommended plots

```
df ['pandas_SMA_3'] = df.iloc[:,1].rolling(window=3).mean()

df.head()
```

| | month | demand | pandas_SMA_3 |
|---|---|---|---|
| **0** | 1 | 290 | NaN |
| **1** | 2 | 260 | NaN |
| **2** | 3 | 288 | 279.333333 |
| **3** | 4 | 300 | 282.666667 |
| **4** | 5 | 310 | 299.333333 |

Next steps: View recommended plots

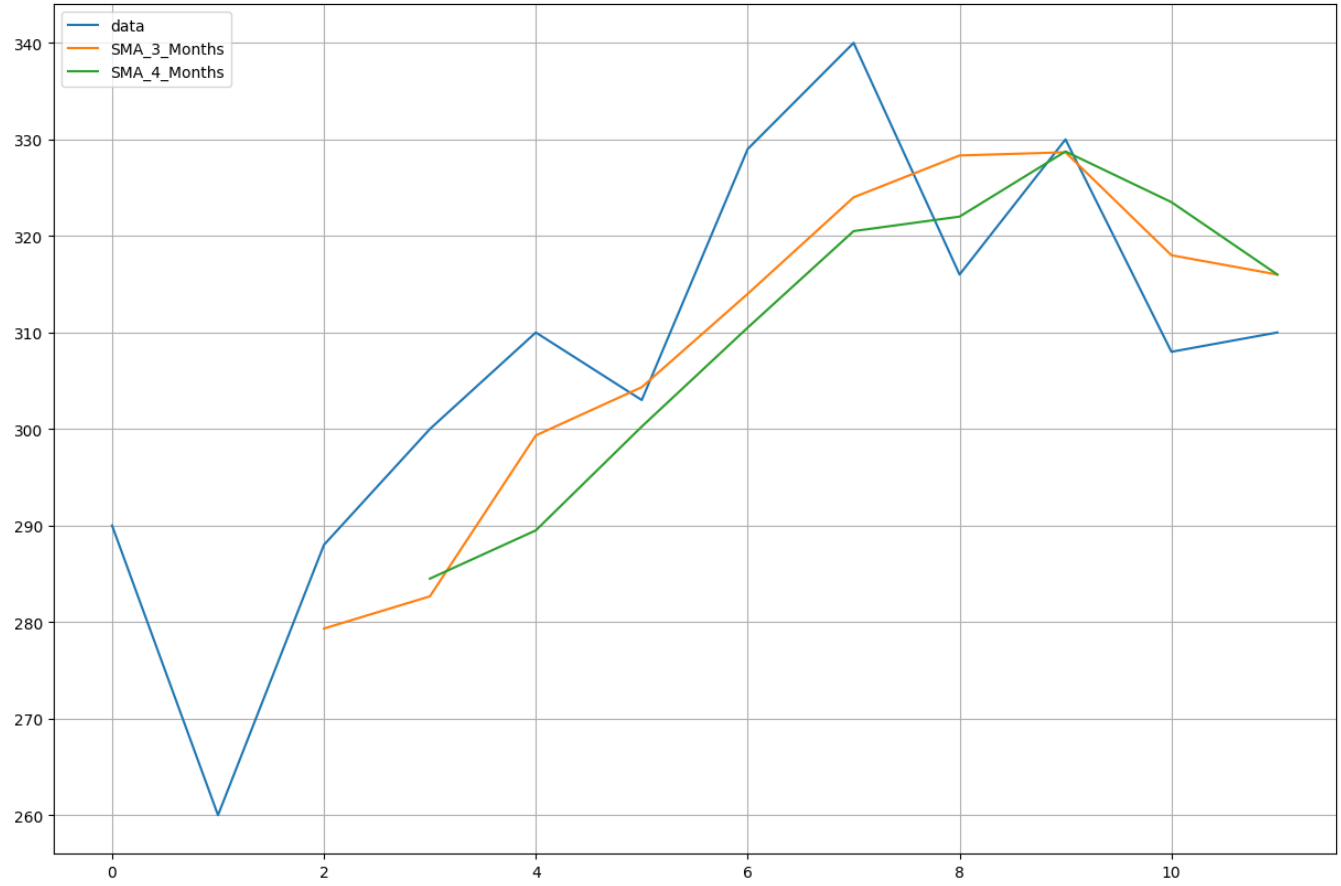
```
df ['pandas_SMA_4'] = df.iloc[:,1].rolling(window=4).mean()

df.head()
```

| | month | demand | pandas_SMA_3 | pandas_SMA_4 |
|---|---|---|---|---|
| 0 | 1 | 290 | NaN | NaN |
| 1 | 2 | 260 | NaN | NaN |

Next steps: View recommended plots

```
plt.figure(figsize=[15,10])
plt.grid(True)
plt.plot(df["demand"],label='data')
plt.plot(df['pandas_SMA_3'], label='SMA_3_Months')
plt.plot(df['pandas_SMA_4'],label='SMA_4_Months')
plt.legend(loc=2)
```

```
<matplotlib.legend.Legend at 0x786a5414a200>
```




**Conclusion**

Time series decomposition is one of the best ways to understand how a time series behaves. The statsmodels library provides an implementation of the naive, or classical, decomposition method in a function called seasonal_decompose().