

Onions Production with Time

Pakistan - India - Bangladesh

1. Importing Libraries and Datafile

```
In [ ]: # Importing Libraries

import seaborn as sns
import pandas as pd
import matplotlib.pyplot as plt
import plotly.express as px

# Loading csv file
df = pd.read_csv('Onion_Production.csv')           #< Write the name of the csv file to import as filename.csv
```

2. Analyzing Dataset

```
In [ ]: #Pulling the header of the dataset
df.head(10)
```

| | Domain Code | Domain | Area Code (FAO) | Country | Element Code | Element | Item Code (FAO) | Item | Year Code | Year | Unit | Value | Flag | Flag Description |
|---|-------------|------------------------------|-----------------|------------|--------------|------------|-----------------|-------------|-----------|------|--------|--------|------|------------------|
| 0 | QCL | Crops and livestock products | 16 | Bangladesh | 5510 | Production | 403 | Onions, dry | 1961 | 1961 | tonnes | 136500 | NaN | Official data |
| 1 | QCL | Crops and livestock products | 16 | Bangladesh | 5510 | Production | 403 | Onions, dry | 1962 | 1962 | tonnes | 130600 | NaN | Official data |
| 2 | QCL | Crops and livestock products | 16 | Bangladesh | 5510 | Production | 403 | Onions, dry | 1963 | 1963 | tonnes | 140300 | NaN | Official data |
| 3 | QCL | Crops and livestock products | 16 | Bangladesh | 5510 | Production | 403 | Onions, dry | 1964 | 1964 | tonnes | 132100 | NaN | Official data |
| 4 | QCL | Crops and livestock products | 16 | Bangladesh | 5510 | Production | 403 | Onions, dry | 1965 | 1965 | tonnes | 145200 | NaN | Official data |
| 5 | QCL | Crops and livestock products | 16 | Bangladesh | 5510 | Production | 403 | Onions, dry | 1966 | 1966 | tonnes | 150400 | NaN | Official data |
| 6 | QCL | Crops and livestock products | 16 | Bangladesh | 5510 | Production | 403 | Onions, dry | 1967 | 1967 | tonnes | 198100 | NaN | Official data |
| 7 | QCL | Crops and livestock products | 16 | Bangladesh | 5510 | Production | 403 | Onions, dry | 1968 | 1968 | tonnes | 198100 | NaN | Official data |
| 8 | QCL | Crops and livestock products | 16 | Bangladesh | 5510 | Production | 403 | Onions, dry | 1969 | 1969 | tonnes | 185900 | NaN | Official data |
| 9 | QCL | Crops and livestock products | 16 | Bangladesh | 5510 | Production | 403 | Onions, dry | 1970 | 1970 | tonnes | 181900 | NaN | Official data |

```
In [ ]: # Quick Stats of the Dataset
df.describe()
```

| | Area Code (FAO) | Element Code | Item Code (FAO) | Year Code | Year | Value |
|-------|-----------------|--------------|-----------------|-------------|-------------|--------------|
| count | 180.000000 | 180.0 | 180.0 | 180.000000 | 180.000000 | 1.800000e+02 |
| mean | 93.666667 | 5510.0 | 403.0 | 1990.500000 | 1990.500000 | 2.753572e+06 |
| std | 61.163760 | 0.0 | 0.0 | 17.366409 | 17.366409 | 5.064340e+06 |
| min | 16.000000 | 5510.0 | 403.0 | 1961.000000 | 1961.000000 | 9.508800e+04 |
| 25% | 16.000000 | 5510.0 | 403.0 | 1975.750000 | 1975.750000 | 1.856500e+05 |
| 50% | 100.000000 | 5510.0 | 403.0 | 1990.500000 | 1990.500000 | 1.148748e+06 |
| 75% | 165.000000 | 5510.0 | 403.0 | 2005.250000 | 2005.250000 | 2.120256e+06 |
| max | 165.000000 | 5510.0 | 403.0 | 2020.000000 | 2020.000000 | 2.673800e+07 |

3. Defining Components of the Graph

```
In [ ]: # Defining the main data axis and components of the graph

xAX = 'Year'           #< Select the data column for your x-axis
yAX = 'Value'          #< Select the data column for your y-axis
snshue = 'Country'     #< Select the data column for your hue option

# Defining Graph Title, axis lables
LTitle = 'Onions Production wrt Countries'           #< Write your Chart Title
xLabel = 'Year'                                     #< Write your x-axis lable
yLabel = 'Weight in 10 MM (tonnes)'                  #< Write your y-axis lable
LegTitle = 'Countries'                             #< Write your Legend Title
```

4. Plotting Varioius Graphs as Options

```
In [ ]: #Line Plot of the Data using ploty
fig = px.line(df, x = xAX, y = yAX, color =snshue, title= LTitle)

# Defining Titles and lables
fig.update_layout(
    title=LTitle,
    xaxis_title=xLabel,
    yaxis_title=yLabel,
    legend_title=LegTitle,
)
#Slider
fig.update_xaxes(rangeslider_visible=True)

fig.show()
```

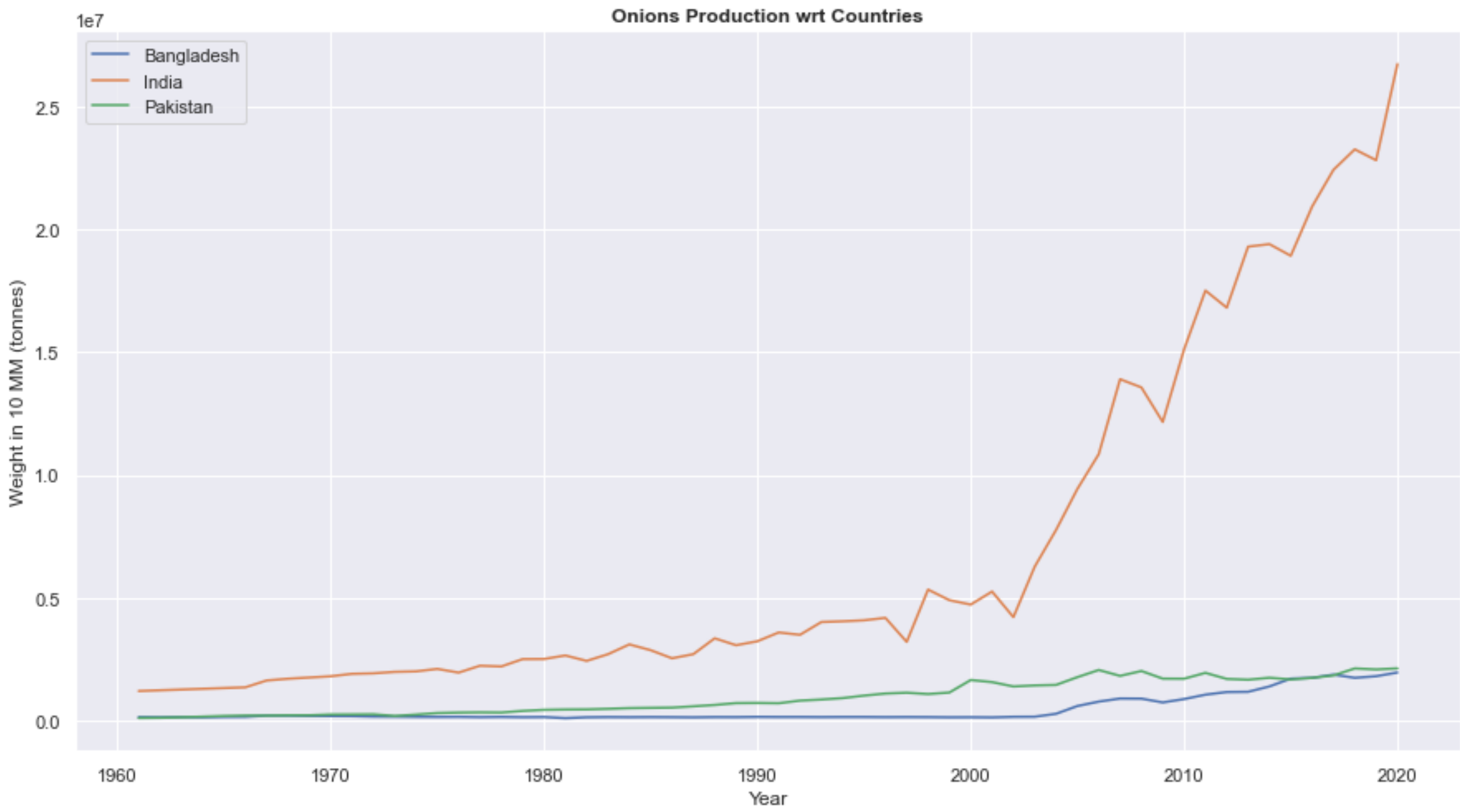
```
In [ ]: #Draww a line plot using Seaborn

#Defining Graph size
sns.set(rc = {'figure.figsize':(15,8)})

#Plot the graph
sns.lineplot(x=xAX, y=yAX, hue=snshue, data=df, ci=95)

# Defining Titles and lables
plt.title(LTitle, size="12", weight="bold")
plt.xlabel(xLabel)
plt.ylabel(yLabel)
plt.legend()

plt.show()
```



```
In [ ]: #Draww a lm plot using Seaborn

# Setting Theme
sns.set_theme(style="darkgrid")

# Make a custom palette with Country colors
pal = dict(Pakistan="green", Bangladesh="blue", India="orange")

# time plot based on Value(Area) and Country
g = sns.lmplot(x=xAX, y=yAX, hue=snshue, data=df, palette=pal, y_jitter=10, logistic=False, truncate=False, height=10)

# Defining Titles and lables
plt.title(LTitle, size="12", weight="bold")
plt.xlabel(xLabel)
plt.ylabel(yLabel)
plt.legend()
plt.ylim()

plt.show()
```

