Onions Harvested 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_Area_Harvested.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)
```

Out[ ]:

	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	5312	Area harvested	403	Onions, dry	1961	1961	ha	22300	NaN	Official data
1	QCL	Crops and livestock products	16	Bangladesh	5312	Area harvested	403	Onions, dry	1962	1962	ha	23500	NaN	Official data
2	QCL	Crops and livestock products	16	Bangladesh	5312	Area harvested	403	Onions, dry	1963	1963	ha	25500	NaN	Official data
3	QCL	Crops and livestock products	16	Bangladesh	5312	Area harvested	403	Onions, dry	1964	1964	ha	25500	NaN	Official data
4	QCL	Crops and livestock products	16	Bangladesh	5312	Area harvested	403	Onions, dry	1965	1965	ha	27500	NaN	Official data
5	QCL	Crops and livestock products	16	Bangladesh	5312	Area harvested	403	Onions, dry	1966	1966	ha	28300	NaN	Official data
6	QCL	Crops and livestock products	16	Bangladesh	5312	Area harvested	403	Onions, dry	1967	1967	ha	34400	NaN	Official data
7	QCL	Crops and livestock products	16	Bangladesh	5312	Area harvested	403	Onions, dry	1968	1968	ha	34400	NaN	Official data
8	QCL	Crops and livestock products	16	Bangladesh	5312	Area harvested	403	Onions, dry	1969	1969	ha	33200	NaN	Official data
9	QCL	Crops and livestock products	16	Bangladesh	5312	Area harvested	403	Onions, dry	1970	1970	ha	32800	NaN	Official data

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

Out[ ]:

	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	5312.0	403.0	1990.500000	1990.500000	2.083514e+05
std	61.163760	0.0	0.0	17.366409	17.366409	3.034414e+05
min	16.000000	5312.0	403.0	1961.000000	1961.000000	1.210000e+04
25%	16.000000	5312.0	403.0	1975.750000	1975.750000	3.399850e+04
50%	100.000000	5312.0	403.0	1990.500000	1990.500000	1.127505e+05
75%	165.000000	5312.0	403.0	2005.250000	2005.250000	1.862762e+05
max	165.000000	5312.0	403.0	2020.000000	2020.000000	1.434000e+06

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 = 'Area Harvested with Onion wrt Countries'          #< Write your Chart Title
xLable = 'Year'                                             #< Write your x-axis lable
yLable = 'Area in MM (ha)'                                  #< Write your y-axis lable
LegTitle = 'Countries'                                      #< Write your Legend Title
```

4. Plotting Varioius Graphs as Opitions

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=xLable,
    yaxis_title=yLable,
    legend_title=LegTitle,
)
#Slider
fig.update_xaxes(rangeslider_visible=True)

fig.show()
```

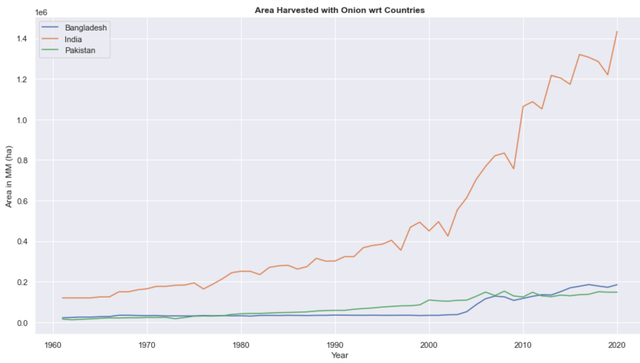
In [ ]:
```
#Drarw a line plot using Seaborn

#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(xLable)
plt.ylabel(yLable)
plt.legend()

plt.show()
```



In [ ]:
```
#Drarw 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(xLable)
plt.ylabel(yLable)
plt.legend()
plt.ylim()

plt.show()
```