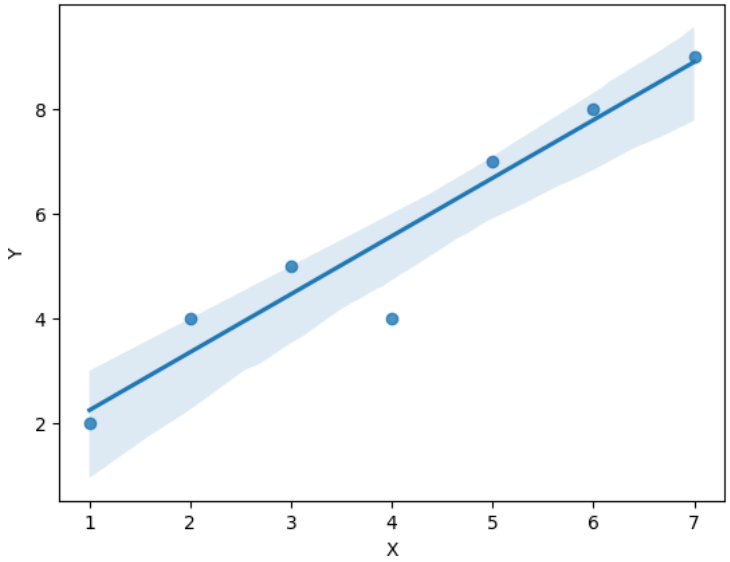
Name:-Prajakta Yuvraj Koli

Practical:-9

Roll no:-24 Sub:-DV

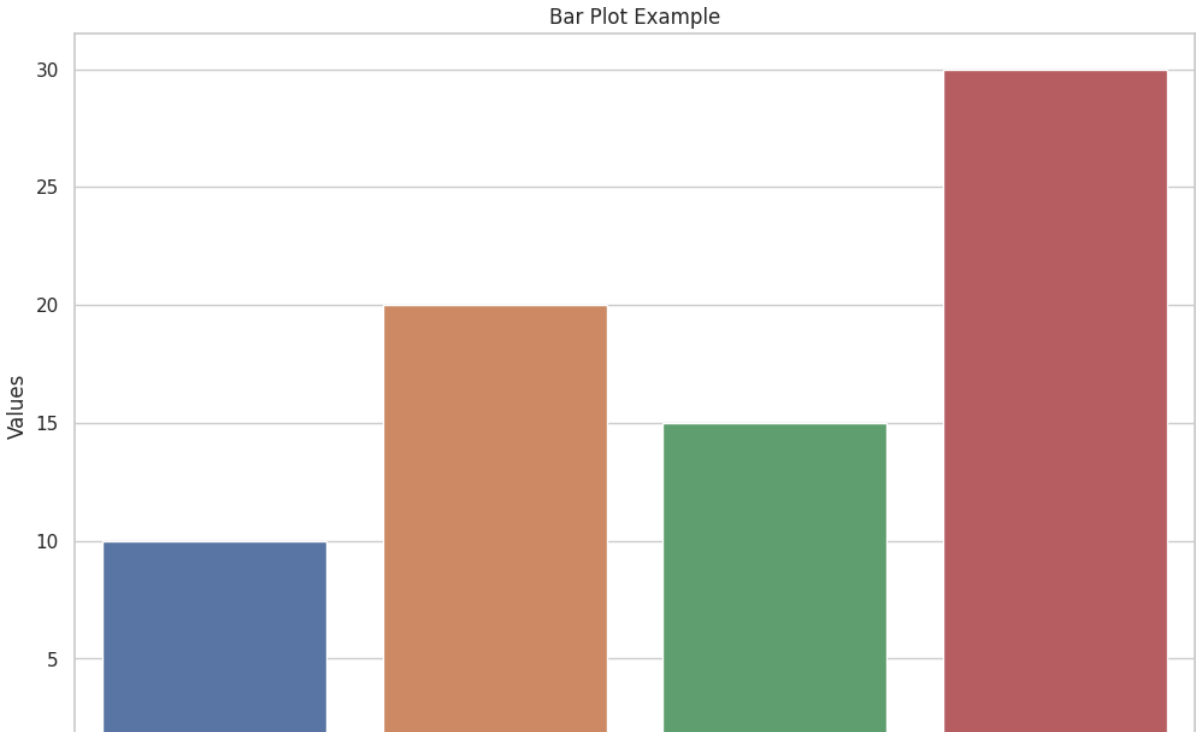
```
import seaborn as sns
import matplotlib.pyplot as plt

# Sample data
import pandas as pd
data = pd.DataFrame({
    'X': [1, 2, 3, 4, 5, 6, 7],
    'Y': [2, 4, 5, 4, 7, 8, 9]
})

# Create a regression plot using regplot
sns.regplot(x='X', y='Y', data=data)
```

```
<Axes: xlabel='X', ylabel='Y'>
```

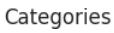


barplot

```
import seaborn as sns


# Sample data
categories = ['Category A', 'Category B', 'Category C', 'Category D']
values = [10, 20, 15, 30]

# Create a bar plot using Seaborn

plt.figure(figsize=(12, 8)) # Set the figure size

# Create the bar plot
sns.barplot(x=categories, y=values)

# Add labels and a title
plt.xlabel('Categories')
plt.ylabel('Values')
plt.title('Bar Plot Example')

# Show the plot
plt.show()
```

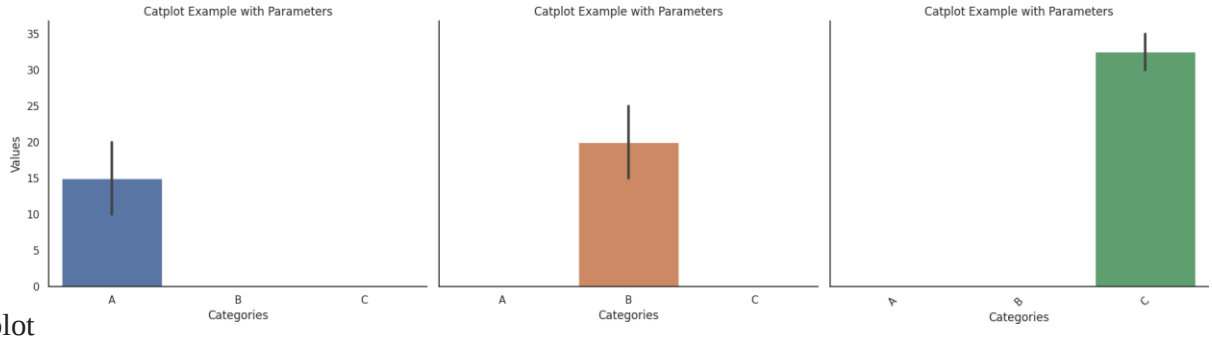


catplplot




```
import seaborn as sns


# Sample data in a DataFrame
import pandas as pd
data = pd.DataFrame({'Category': ['A', 'B', 'A', 'B', 'C', 'C'],
                     'Value': [10, 15, 20, 25, 30, 35]})

# Create a catplot with various parameters
sns.set(style="white")  # Set the style for the plot
g = sns.catplot(x="Category", y="Value", kind="bar", data=data,
                hue="Category", col="Category", dodge=False, aspect=1.2)

  # kind="box": This specifies the kind of plot, which is a bar plot in this case.
#  hue="Category": We use the "Category" column for coloring different categories.
 #col="Category": We create subplots for each category.
#aspect=1.2: Sets the aspect ratio of the plot.
#plt.xticks(rotation=45): Rotates the x-axis labels by 45 degrees for better readability.
#dodge=False: This parameter prevents the boxes from being dodged when using the "hue" parameter.
#

# Customize the plot (labels, title, etc.)
g.set_axis_labels("Categories", "Values")
g.set(title="Catplot Example with Parameters")

# Additional customization
plt.xticks(rotation=45) # Rotate x-axis labels for better readability

# Show the plot
plt.show()
```

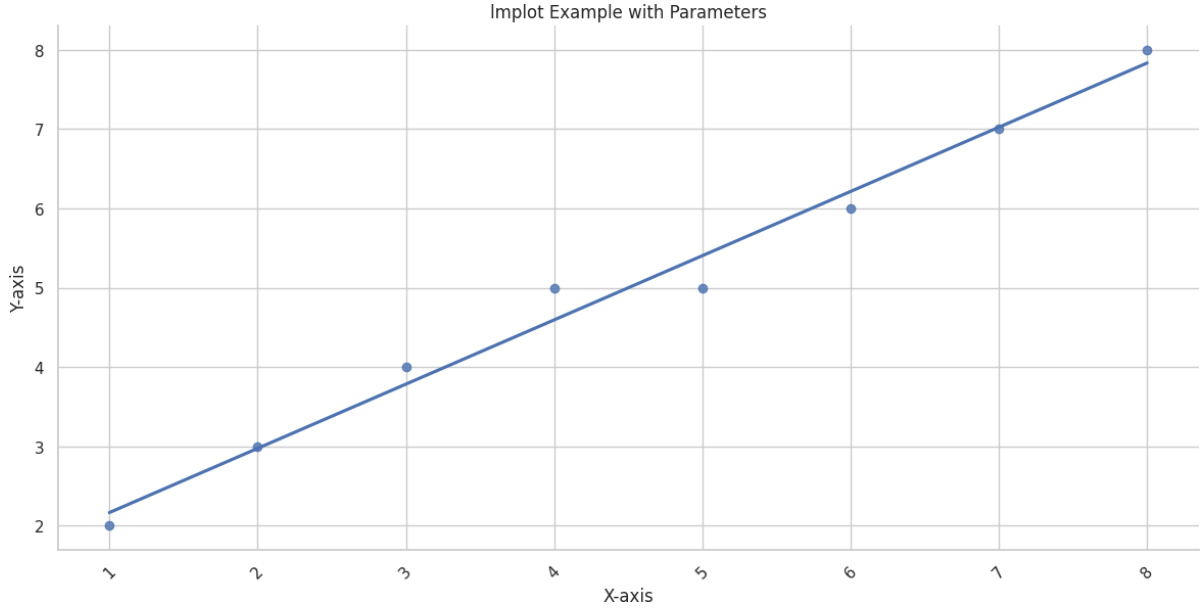
```
C:\Users\HP\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```



implot

```
import seaborn as sns
import matplotlib.pyplot as plt

# Sample data in a DataFrame
import pandas as pd
data = pd.DataFrame({'X': [1, 2, 3, 4, 5, 6, 7, 8],
                     'Y': [2, 3, 4, 5, 5, 6, 7, 8]})

# Create an lmplot with various parameters
sns.set(style="whitegrid")  # Set the style for the plot
g = sns.lmplot(x="X", y="Y", data=data, aspect=2, height=6, ci=None)

# Customize the plot (labels, title, etc.)
g.set_axis_labels("X-axis", "Y-axis")
g.set(title="lmplot Example with Parameters")

# Additional customization
plt.xticks(rotation=45) # Rotate x-axis labels for better readability

# Show the plot
plt.show()
```

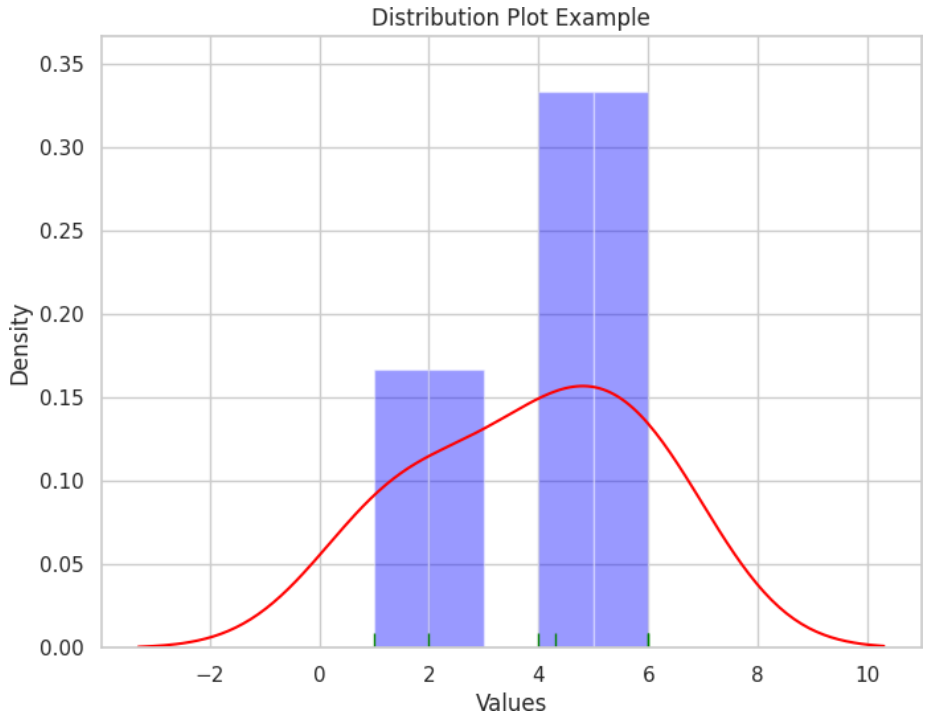
```
C:\Users\HP\AppData\Local\Programs\Python\Python310\lib\site-packages\seaborn\axisgrid.py:118: UserWarning: The figure layout has changed to tight
  self._figure.tight_layout(*args, **kwargs)
```



diplot

```
import seaborn as sns
import matplotlib.pyplot as plt

# Sample data
data = [4,6,2,6,1,4.3]

# Create a distribution plot
sns.set(style="whitegrid") # Set the style for the plot
plt.figure(figsize=(8, 6)) # Set the figure size

# Create the distplot
sns.distplot(data, bins=5, hist=True, kde=True, rug=True, hist_kws={'color': 'blue'}, kde_kws={'color': 'red'}, rug_kws={'color': 'green'})
#bins (optional): This parameter controls the number of bins or intervals in the histogram.
#hist (optional): If True, it displays a histogram. You can set it to False if you only want the KDE plot. By default, it is set to True.
#kde (optional): If True, it overlays a KDE plot on top of the histogram. You can set it to False if you only want the histogram. By default, it is set to True.
#rug (optional): If True, it adds small vertical tick marks (rug plot) along the x-axis to show the data points. By default, it is set to False.
#hist_kws (optional): A dictionary of keyword arguments that can be used to customize the appearance of the histogram. For example, you can set the color of the bars, th
#kde_kws (optional): A dictionary of keyword arguments for customizing the appearance of the KDE plot.
#rug_kws (optional): A dictionary of keyword arguments for customizing the appearance of the rug plot.

# Add labels and a title
plt.xlabel('Values')
plt.ylabel('Density')
plt.title('Distribution Plot Example')

# Show the plot
plt.show()
```

```
C:\Users\HP\AppData\Local\Temp\ipykernel_18136\3287290666.py:12: UserWarning:

`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with
similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see
https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751

  sns.distplot(data, bins=5, hist=True, kde=True, rug=True, hist_kws={'color': 'blue'}, kde_kws={'color': 'red'}, rug_kws={'color': 'green'})
```



#

```
import seaborn as sns
import matplotlib.pyplot as plt

# Sample data
data = [2,5, 4 , 6,3,2,5,6,4]

# Create a KDE plot
plt.figure(figsize=(8, 6)) # Set the figure size

# Create the KDE plot
sns.kdeplot(data, shade=True, color='blue', label='KDE Plot')

# Add labels and a title
plt.xlabel('Values')
plt.ylabel('Density')
plt.title('KDE Plot Example')

# Show the legend
plt.legend()

# Show the plot
plt.show()
```

```
C:\Users\HP\AppData\Local\Temp\ipykernel_18136\3213133653.py:11: FutureWarning:

`shade` is now deprecated in favor of `fill`; setting `fill=True`.
This will become an error in seaborn v0.14.0; please update your code.

  sns.kdeplot(data, shade=True, color='blue', label='KDE Plot')
```