Training Day – 32

Topic:* Bar Charts and Histograms in Matplotlib
 - Created bar charts and histograms to visualize data distributions.
 - Example: Plotted a histogram for a dataset of random integers.

opic: Bar Charts and Histograms in Matplotlib

- Visualizing categorical and continuous data distributions.

- Example:

```
data = [5, 7, 8, 6, 7]
plt.bar(range(len(data)), data)
plt.hist(data, bins=3)
plt.show()
```

•**In "barplot"** some attributes are practiced to visualize the given data are as follows-
     -> color: It help. to change the colour of bars in graph.
     -> width: It is used to change the broadness of bars in
     in the graph.
     ->.xlabel: It used for naming of the X-axis of the
     bar graph.
     ->.ylabel: It used for naming of the Y-axis of the
     bar graph.
     ->.barh: This attribute change the direction of bars,
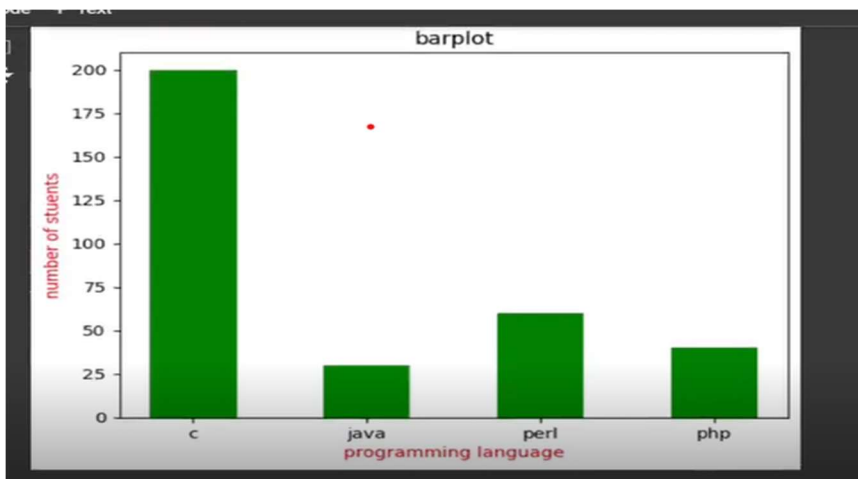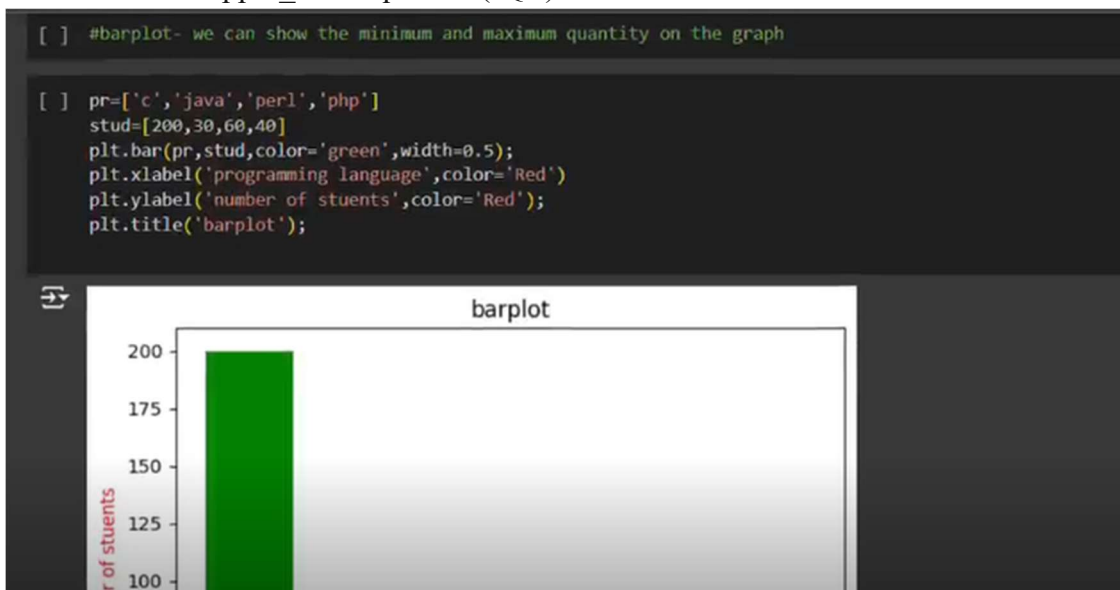   i.e. Vertical to Horizontal.

   •As same like "bar graph" attributes are same in "histogram".
   •In " piechart" some attributes are used to help in the visualization of given data are -
    ->explode: It help to point out the selected portion of
    the pie-chart by exploding the part from
    the whole chart according to given
    explode value.
    ->autopct: It represent the data in percentage
    automatically and it's value is-
    "autopct='%1.2f%%' ".

   •In box plot there are some concepts which are important to understand for making
   visualization easier and identifying outliers easily.

1.IQR: It stand for "inter quartile range", which define as the difference of "third quartile(q3) and first quartile (q0)".

2. Outliers are those value which comes after the last quartile to affect our mean, as well as below the first quartile.

3. Our whole data is divided in four part i.e. 25%, 50%, 75%, 100%, and these percentile values refers to our quartile(q1,q2,q3,q4).

4.The value of lower_limit is extracted by applying
 formula of lower_limit = q1-1. 5*(IQR).
 The value of upper_limit is extracted by applying
 formula of upper_limit=q3+1.5*(IQR).

```
[ ]  #barplot- we can show the minimum and maximum quantity on the graph
```

```python
[ ]  pr=['c','java','perl','php']
     stud=[200,30,60,40]
     plt.bar(pr,stud,color='green',width=0.5);
     plt.xlabel('programming language',color='Red')
     plt.ylabel('number of stuents',color='Red');
     plt.title('barplot');
```

Training Day – 33

Topic:* Scatter Plots in Matplotlib

 - Plotted relationships between two variables.

 - Example: Visualized correlation between "Age" and "Height" columns.
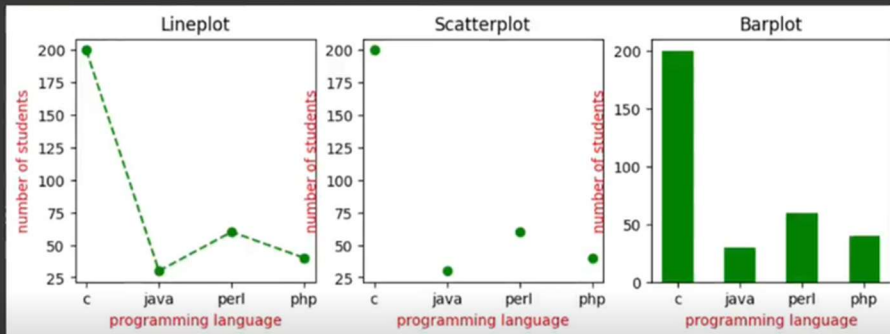
- **Example:**

```python
x = [1, 2, 3]
y = [2, 4, 1]
plt.scatter(x, y)
plt.title("Scatter Plot")
plt.show()
```

visualization easier and identifying outliers easily.

1. IQR: It stand for "inter quartile range", which define as the difference of "third quartile(q3) and first quartile (q0)".
2. Outliers are those value which comes after the last quartile to affect our mean, as well as below the first quartile.
3. Our whole data is divided in four part i.e. 25%, 50%, 75%, 100%, and these percentile values refers to our quartile(q1,q2,q3,q4).
4. The value of lower_limit is extracted by applying
   formula of lower_limit = q1-1. 5*(IQR).
   The value of upper_limit is extracted by applying
   formula of upper_limit=q3+1.5*(IQR).

```python
pr=['c','java','perl','php']
stud=[200,30,60,40]
plt.subplot(1,3,2)
plt.scatter(pr,stud,color='green');
plt.xlabel('programming language',color='Red')
plt.ylabel('number of students',color='Red');
plt.title('Scatterplot');

pr=['c','java','perl','php']
stud=[200,30,60,40]
plt.subplot(1,3,3)
plt.bar(pr,stud,color='green',width=0.5);
plt.xlabel('programming language',color='Red')
plt.ylabel('number of students',color='Red');
plt.title('Barplot');
```

```
df=sns.load_dataset('tips')
df
```

	total_bill	tip	sex	smoker	day	time	size
0	16.99	1.01	Female	No	Sun	Dinner	2

1:45   1x

Training Day – 34

Topic:* Stacked Charts with Legends in Matplotlib

 - Created stacked area charts and added legends and titles.
 - Example: Visualized cumulative sales over time for different products.

Stacked bar plots represent different groups on the highest of 1 another. The peak of the bar depends on the resulting height of the mixture of the results of the groups. It goes from rock bottom to the worth rather than going from zero to value.

Topic: Stacked Charts with Legends in Matplotlib

- **Example:**

```python
x = [1, 2, 3]
y1 = [2, 3, 4]
y2 = [3, 4, 5]
plt.stackplot(x, y1, y2, labels=["Group 1", "Group 2"])
plt.legend()
plt.show()
```

visualization easier and identifying outliers easily.

 1.IQR: It stand for "inter quartile range", which define as the difference of "third quartile(q3) and first quartile (q0)".

 2. Outliers are those value which comes after the last quartile to affect our mean, as well as below the first quartile.

 3. Our whole data is divided in four part i.e. 25%, 50%, 75%, 100%, and these percentile values refers to our quartile(q1,q2,q3,q4).

 4.The value of lower_limit is extracted by applying
 formula of lower_limit = q1-1. 5*(IQR).
 The value of upper_limit is extracted by applying
 formula of upper_limit=q3+1.5*(IQR).

EXAMPLE

importing package

import matplotlib.pyplot as plt

create data

x = ['A', 'B', 'C', 'D']
y1 = [10, 20, 10, 30]
y2 = [20, 25, 15, 25]
# plot bars in stack manner
plt.bar(x, y1, color='r')

```
plt.bar(x, y2, bottom=y1, color='b')
plt.show()
```
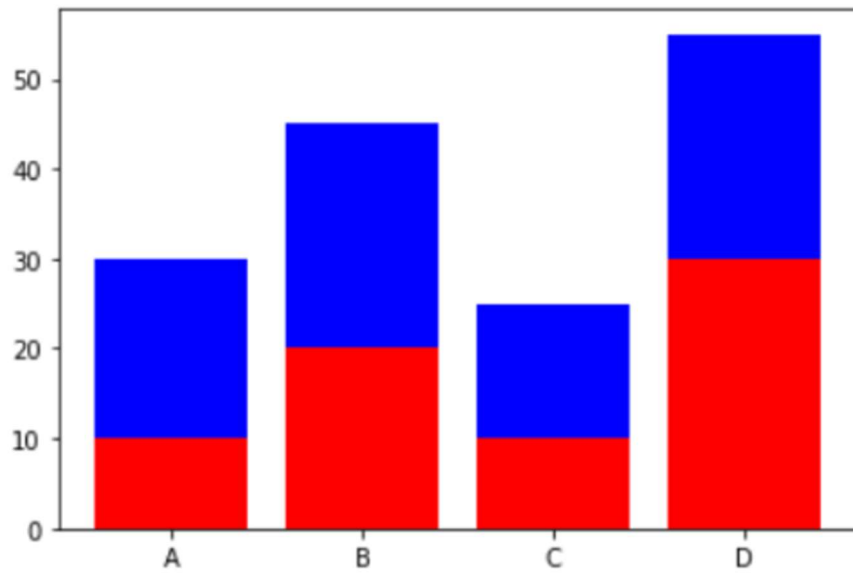
Training Day – 30

Topic: Pandas Groupby Operations

- Grouped data by specific columns and applied aggregation functions.
- Example: Grouped by "Category" column and calculated sums for each group. we will use the *groupby()* function to perform various operations on grouped data.

```python
df.groupby(['Type 1'])['Attack'].max().sort_values(ascending=False)
```

```
Type 1
Psychic     190
Bug         185
Dragon      180
Ground      180
Ghost       165
Rock        165
Normal      160
Fire        160
Water       155
Dark        150
Steel       150
Fighting    145
Grass       132
Fairy       131
Ice         130
Electric    123
Flying      115
Poison      106
Name: Attack, dtype: int64
```

```python
df.groupby(['Type 1'])['Attack'].agg(['sum','min','max','count','mean']).sort_values(by='max',ascending=False).head(1)
```

```python
df['Name'].nunique()
```

```
800
```

```python
df.groupby(['Name'])['Attack'].sum()
```

```
Name
Abomasnow                 92
AbomasnowMega Abomasnow   132
Abra                      20
Absol                     130
AbsolMega Absol           150
                         ...
Zoroark                   105
Zorua                     65
Zubat                     45
Zweilous                  85
Zygarde50% Forme          100
Name: Attack, Length: 800, dtype: int64
```

Training Day – 31

Topic:* Introduction to Matplotlib

 - Learned to create simple line plots.

 - Example: Plotted y = x^2 for x in range 0 to 10.

Matplotlib is a powerful and versatile open-source plotting library for Python, designed to help users visualize data in a variety of formats. Developed by John D. Hunter in 2003, it enables users to graphically represent data, facilitating easier analysis and understanding. **If you want to convert your boring data into interactive plots and graphs, Matplotlib is the tool for you**.

```
1   import matplotlib.pyplot as plt
2
3   x = [0, 1, 2, 3, 4]
4   y = [0, 1, 4, 9, 16]
5
6   plt.plot(x, y)
7   plt.show()
```

ıt:

Simplest plot in Matplotlib
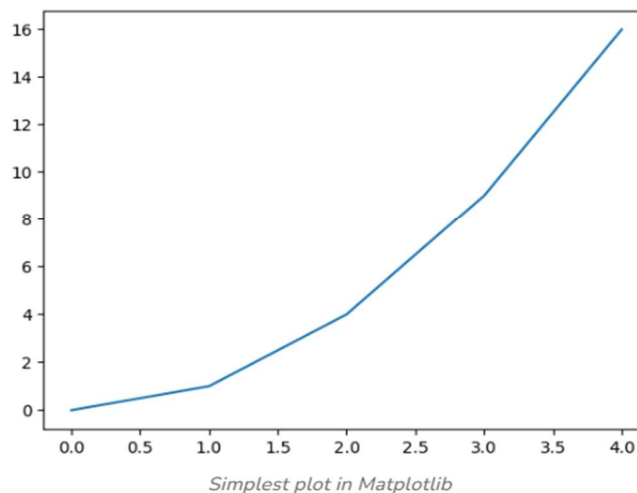
- **Import Matplotlib:** Start by importing matplotlib.pyplot as plt.
- **Create Data:** Prepare your data in the form of lists or arrays.
- **Plot Data**: Use plt.plot() to create the plot.
- **Customize Plot**: Add titles, labels, and other elements using methods like plt.title(), plt.xlabel(), and plt.ylabel().
- **Display Plot**: Use plt.show() to display the plot.