Introduction to Matplotlib

Course Code: CPE 031	**Program**: Computer Engineering
Course Title: Visualization and Data Analysis	**Date Performed**: 10/22/24
Section: CPE21S4	**Date Submitted**: 10/22/24
Name: SANTOS, ANDREI R.	**Instructor**: Professor Maria Rizette Sayo

Intended Learning Outcomes (ILO):

By the end of this laboratory session, learners will be able to:
1. Utilize Matplotlib's pyplot interface to create a variety of visualizations, including line plots, scatter plots, histograms, and box plots, demonstrating an understanding of the library's syntax and functionality.

2. Customize visual elements such as titles, labels, and legends to enhance the clarity and aesthetics of their plots, applying best practices in data visualization.

3. Analyze and interpret visual data representations to extract meaningful insights, effectively communicating findings through well-structured graphical presentations.

Part 1: Perform the following codes, and understand the difference between line plot, scatter plot, histogram, bar chart, box plot, and pie chart using matplotlib's pyplot sub-module. **(Provide a screenshot of your output.)**

1. Line Plot

```python
import matplotlib.pyplot as plt

x = [1, 2, 3, 4]
y = [10, 20, 25, 30]
plt.plot(x, y)
plt.title("Line Plot Example")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.show()
```
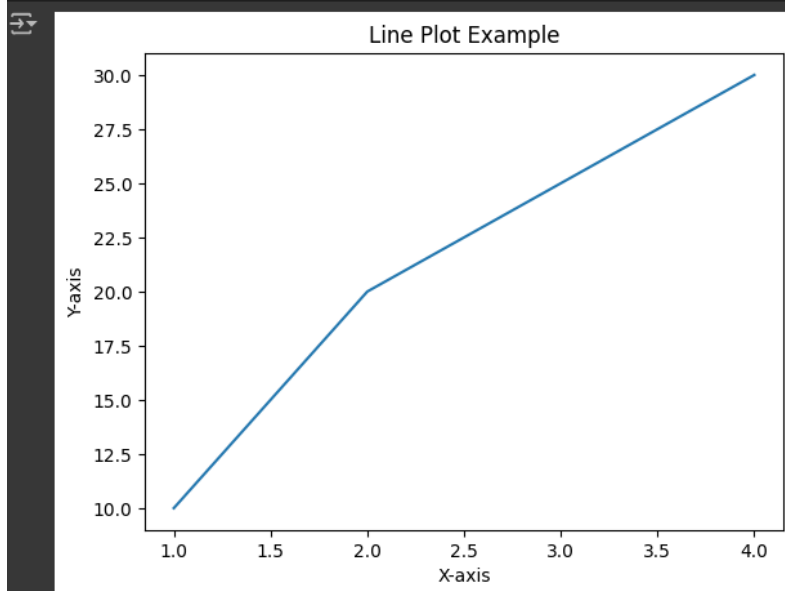
OUTPUT:

2. Scatter Plot

```python
import matplotlib.pyplot as plt

x = [1, 2, 3, 4]
y = [10, 20, 25, 30]
plt.scatter(x, y)
plt.title("Scatter Plot Example")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.show()
```

OUTPUT:

```python
import matplotlib.pyplot as plt

# 2. Scatter Plot

x = [1, 2, 3, 4]
y = [10, 20, 25, 30]
plt.scatter(x,y)
plt.title("Scatter Plot Example")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.show()
```
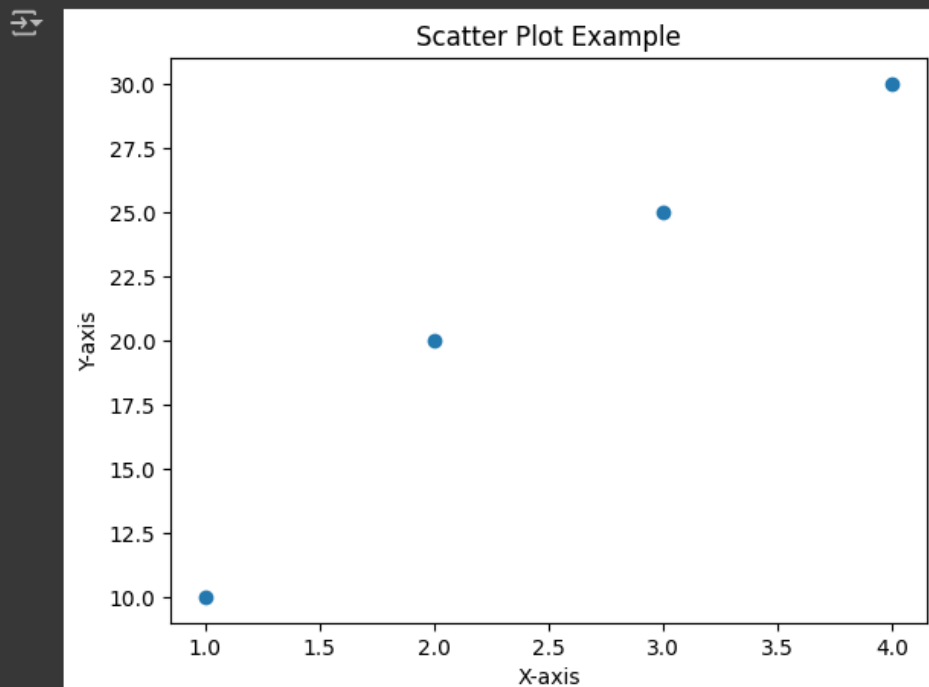
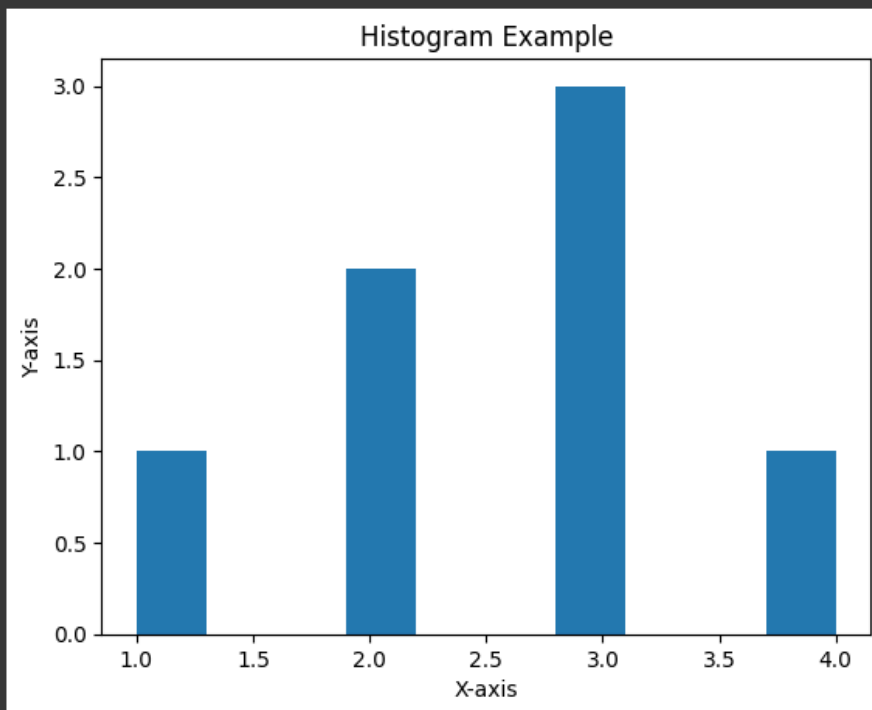3. Histogram

```python
import matplotlib.pyplot as plt

data = [1, 2, 2, 3, 3, 3, 4]
plt.hist(data)
plt.title("Histogram Example")
plt.xlabel("Value")
plt.ylabel("Frequency")
plt.show()
```

OUTPUT:

```python
[133] import matplotlib.pyplot as plt

# 3. Histogram

data = [1, 2, 2, 3, 3, 3, 4]
plt.hist(data)
plt.title("Histogram Example")
plt.xlabel("X-axis")
plt.ylabel("Y-axis")
plt.show()
```

4. Bar Chart

```python
import matplotlib.pyplot as plt

categories = ['A', 'B', 'C']
values = [5, 7, 3]
plt.bar(categories, values)
plt.title("Bar Chart Example")
plt.xlabel("Categories")
plt.ylabel("Values")
plt.show()
```

OUTPUT:

```python
import matplotlib.pyplot as plt

# 4. Bar Chart

categories = ['A', 'B', 'C']
values = [5, 7, 3]
plt.bar(categories, values)
plt.title("Bar Chart Example")
plt.xlabel("Categories")
plt.ylabel("Values")
plt.show()
```
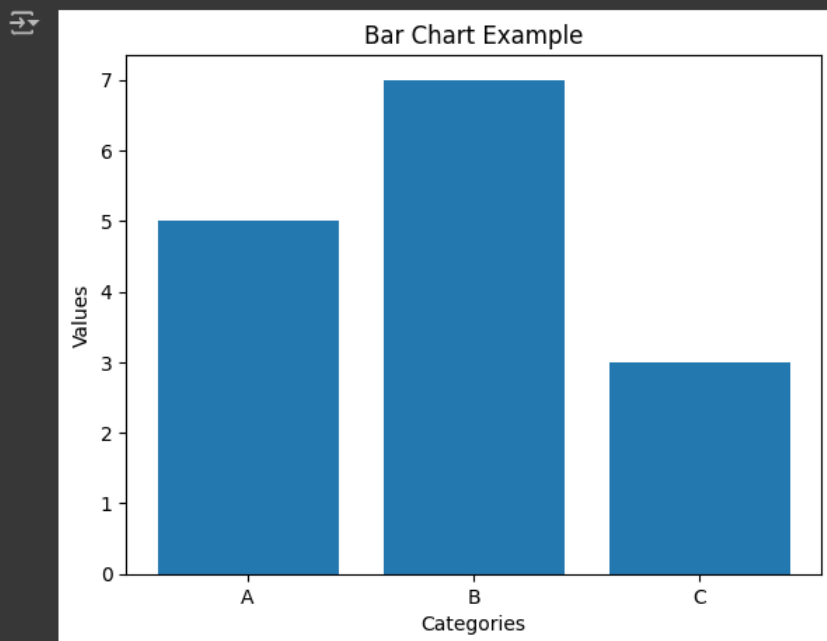
5. Box plot

```python
import matplotlib.pyplot as plt

data = [[1.5]*10 + [2]*10 + [3]*10]

plt.boxplot(data)

plt.title("Box Plot Example")
plt.ylabel("Values")
plt.show()
```

OUTPUT:

```python
import matplotlib.pyplot as plt

# 5. Box Plot

data = [[1.5]*10 + [2.5]*10 + [3]*10]

plt.boxplot(data)
plt.title("Box Plot Example")
plt.xlabel("Values")
plt.show()
```
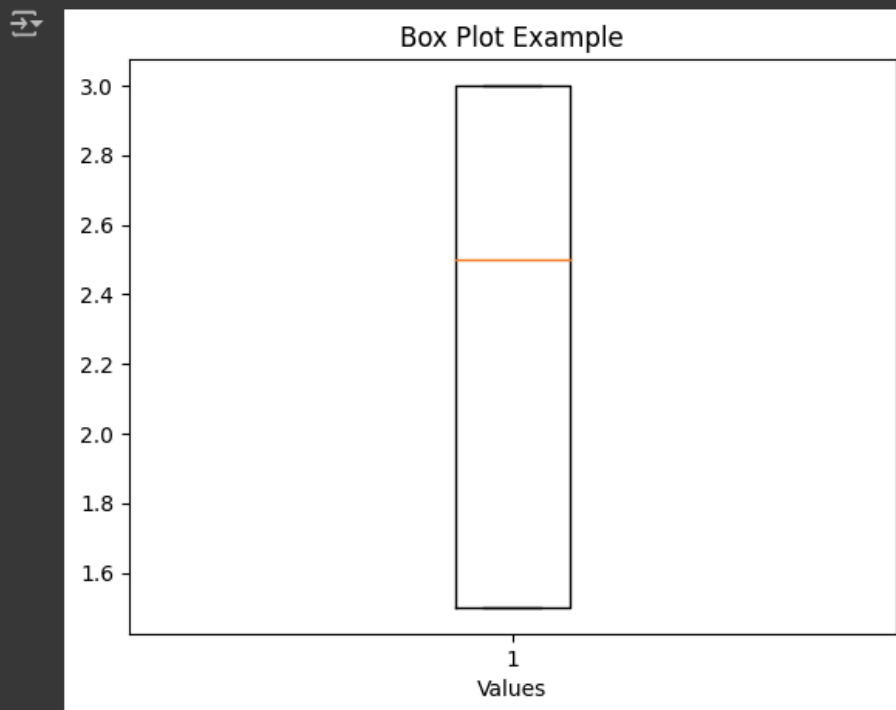
6. Pie chart

```python
import matplotlib.pyplot as plt

labels = ['A', 'B', 'C']
sizes = [40, 30, 30]

plt.pie(sizes, labels=labels)
plt.title("Pie Chart Example")
plt.show()
```

OUTPUT:

```python
import matplotlib.pyplot as plt

# 6. Pie Chart

labels = ['A', 'B', 'C']
sizes = [40, 30, 30]
plt.pie(sizes, labels=labels)
plt.title("Pie Chart Example")
plt.show()
```

Part 2: Refer to the instructions below.

1. **Find a dataset for this activity**: Please visit Kaggle and look for a new dataset that would allow you to perform visualization and analysis using matplotlib.

2. **Creating a dataframe from your CSV file:** Once you have successfully loaded your dataset, you need to create a dataframe from your uploaded CSV file
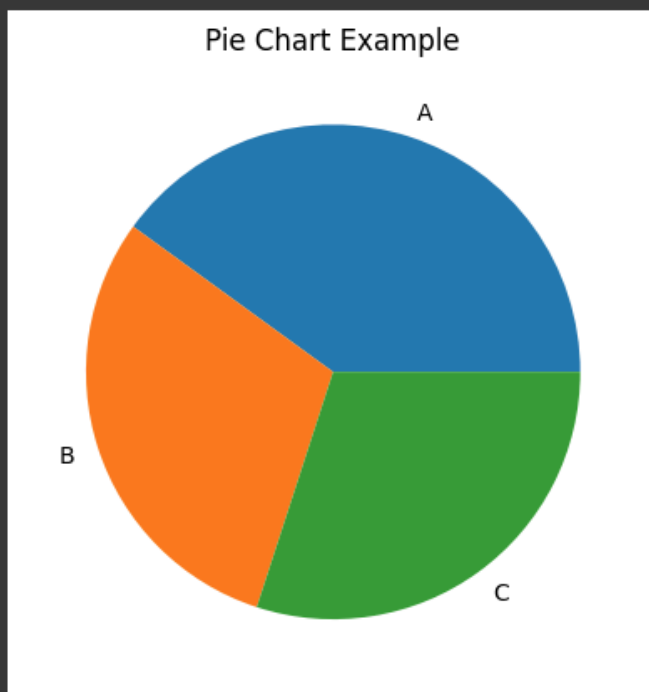
3. **Import the matplotlib.pyplot**

4. **Based on your chosen dataset, you will develop three questions that you will answer using pyplot visualizations. This means that you will need to produce at least three pyplot visualizations. You are also required to make certain customizations on your data vizes.**

5. Provide observations for each of your data viz, then **produce one insight not longer than five sentences given your three observations**. Your output shall follow this outline:
   a. Introduction (Describe your dataset)
   b. Questions
   c. Visualization and Observation
   d. Insight

6. Your grade will depend on the quality of the question, difficulty/complexity of the visualization, and value-add of the insight that you will generate.

OUTPUT:

1. **Find a dataset for this activity**:

- Top 100 most Streamed - Sheet1 (1).csv

2. **Creating a dataframe from your CSV file.**

```
[1] import pandas as pd

    from google.colab import files
    uploaded = files.upload()
```

```
Choose Files  Top 100 mo...eet1 (1).csv
 • Top 100 most Streamed - Sheet1 (1).csv(text/csv) - 7758 bytes, last modified: 10/22/2024 - 100% done
Saving Top 100 most Streamed - Sheet1 (1).csv to Top 100 most Streamed - Sheet1 (1).csv
```

```
data = pd.read_csv('Top 100 most Streamed - Sheet1 (1).csv')
song = pd.DataFrame(data)
print(song)
```

```
                                  title          artist  \
0                        Blinding Lights     The Weeknd
1                       Watermelon Sugar    Harry Styles
2                    Mood (feat. iann dior)      24kGoldn
3                     Someone You Loved   Lewis Capaldi
4                                Perfect     Ed Sheeran
..                                   ...             ...
95                            One Dance           Drake
96                                Sugar       Maroon 5
97                             Emotions     Mark Mendy
98                           Cold Water    Major Lazer
99   I Took A Pill In Ibiza - Seeb Remix    Mike Posner

                     top genre  year  beats.per.minute  energy  danceability  \
0    canadian contemporary r&b  2020               171      73            51
1                          pop  2019                95      82            55
2                     cali rap  2021                91      72            70
3                          pop  2019               110      41            50
4                          pop  2017                95      45            60
..                         ...   ...               ...     ...           ...
95            canadian hip hop  2016               104      61            79
96                         pop  2015               120      79            75
97                   pop dance  2021               126      83            66
98                   dance pop  2018                93      80            61
99                   dance pop  2016               102      73            67
```

```
     loudness.dB  liveness  valance  length  acousticness  speechiness  \
0            -6         9       33     200             0            6
1            -4        34       56     174            12            5
2            -4        32       73     141            17            4
3            -6        11       45     182            75            3
4            -6        11       17     263            16            2
..          ...       ...      ...     ...           ...          ...
95           -6        32       43     174             1            6
96           -7         9       88     235             6            3
97           -5        40       74     172             5           29
98           -5        16       50     185             7            4
99           -7         9       66     198             3           10

    popularity
0           91
1           88
2           88
3           86
4           86
..         ...
95          66
96          66
97          66
98          56
99          53

[100 rows x 14 columns]
```

```python
data = pd.read_csv('Top 100 most Streamed - Sheet1 (1).csv')
song = pd.DataFrame(data)
song.head(50)
```

	title	artist	top genre	year	beats.per.minute	energy	danceability
0	Blinding Lights	The Weeknd	canadian contemporary r&b	2020	171	73	51
1	Watermelon Sugar	Harry Styles	pop	2019	95	82	55
2	Mood (feat. iann dior)	24kGoldn	cali rap	2021	91	72	70
3	Someone You Loved	Lewis Capaldi	pop	2019	110	41	50
4	Perfect	Ed Sheeran	pop	2017	95	45	60
5	Believer	Imagine Dragons	modern rock	2017	125	78	78
6	lovely (with Khalid)	Billie Eilish	electropop	2018	115	30	35
7	Circles	Post Malone	dfw rap	2019	120	76	70
8	Shape of You	Ed Sheeran	pop	2017	96	65	83
9	Memories	Maroon 5	pop	2021	91	33	78
10	Closer	The Chainsmokers	dance pop	2016	95	52	75
11	bad guy	Billie Eilish	electropop	2019	135	43	70
12	Say You Won't Let Go	James Arthur	pop	2016	85	56	36
13	Lucid Dreams	Juice WRLD	chicago rap	2018	84	57	51
14	All of Me	John Legend	neo soul	2013	120	26	42
15	7 rings	Ariana Grande	dance pop	2019	140	32	78
16	Jocelyn Flores	XXXTENTACION	emo rap	2017	134	39	87
17	goosebumps	Travis Scott	rap	2016	130	73	84
18	Wake Me Up	Avicii	dance pop	2013	124	78	53
19	ROCKSTAR (feat. Roddy Ricch)	DaBaby	north carolina hip hop	2020	90	69	75
20	rockstar (feat. 21 Savage)	Post Malone	dfw rap	2018	160	52	59
21	Photograph	Ed Sheeran	pop	2014	108	38	61
22	SAD!	XXXTENTACION	emo rap	2018	75	61	74
23	SICKO MODE	Travis Scott	rap	2018	155	73	83
24	Stressed Out	Twenty One Pilots	modern rock	2015	170	64	73

✓ 0s   completed at 7:43 PM

| loudness.dB	liveness	valance	length	acousticness	speechiness	popularity
-6	9	33	200	0	6	91
-4	34	56	174	12	5	88
-4	32	73	141	17	4	88
-6	11	45	182	75	3	86
-6	11	17	263	16	2	86
-4	8	67	204	6	13	86
-10	10	12	200	93	3	86
-3	9	55	215	19	4	86
-3	9	93	234	58	8	85
-7	8	60	189	84	6	85
-6	11	66	245	41	3	84
-11	10	56	194	33	38	84
-7	9	49	211	70	6	84
-7	34	22	240	35	20	84
-7	13	33	270	92	3	84
-11	9	33	179	59	33	84
-9	30	44	119	47	24	84
-3	15	43	244	8	5	84
-6	16	64	247	0	5	84
-8	10	50	182	25	16	84
-6	13	13	218	12	7	83
-10	10	20	259	61	5	83
-5	12	47	167	26	14	83
-4	12	45	313	1	22	83
-6	6	65	202	5	14	83

3. Import the matplotlib.pyplot

1. Scatter Plot of Danceability vs. Popularity:

```python
import matplotlib.pyplot as plt

data = song.head(50)

# 1. Scatter Plot

x_values = data['danceability']
y_values = data['popularity']
plt.scatter(x_values, y_values, alpha = .5) # I just add this since I like a little transparent color than opaque one.
plt.title("Scatter Plot of Danceability vs Popularity")
plt.xlabel("Danceability")
plt.ylabel("Popularity")
plt.grid()
plt.show()
```
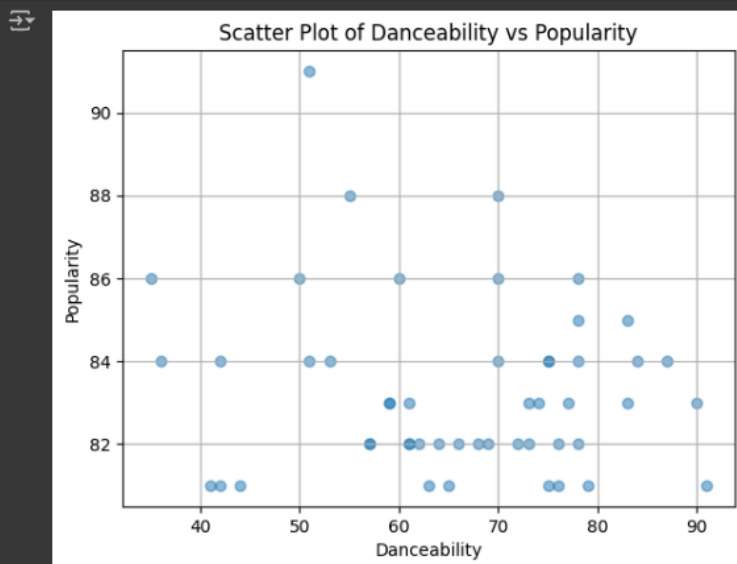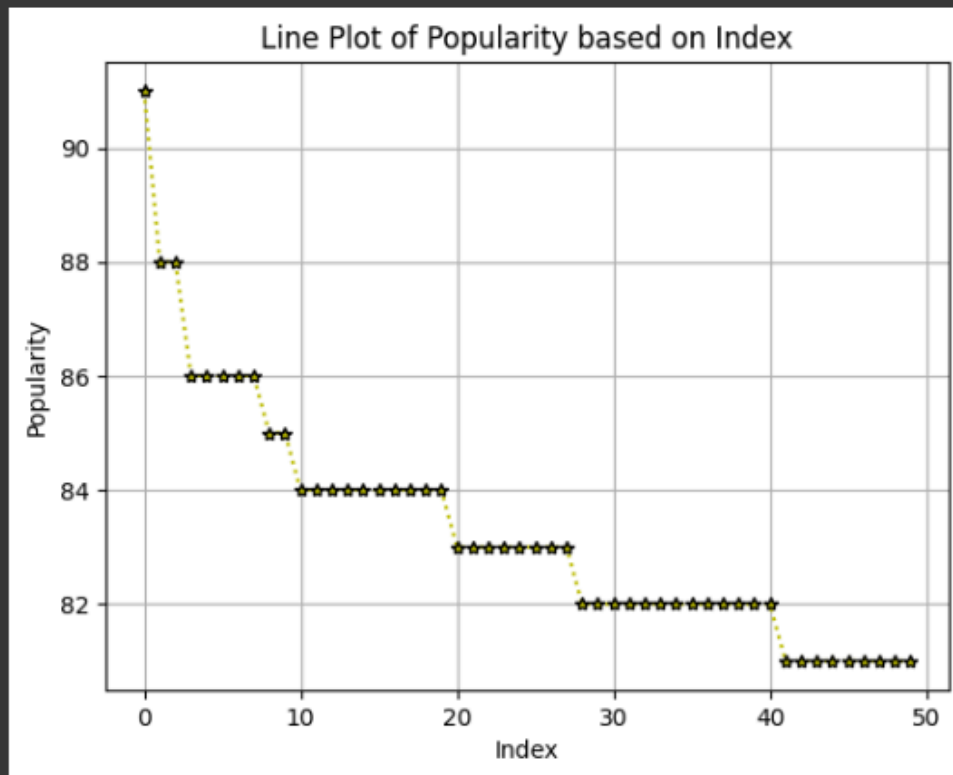
2. Line Plot of Popularity Over Index:

```python
[134] import matplotlib.pyplot as plt

data = song.head(50)

# 2. Line Plot

plt.plot(data['popularity'], '*:y', mec = 'black')
plt.title("Line Plot of Popularity based on Index")
plt.xlabel("Index")
plt.ylabel("Popularity")
plt.grid()
plt.show()
```
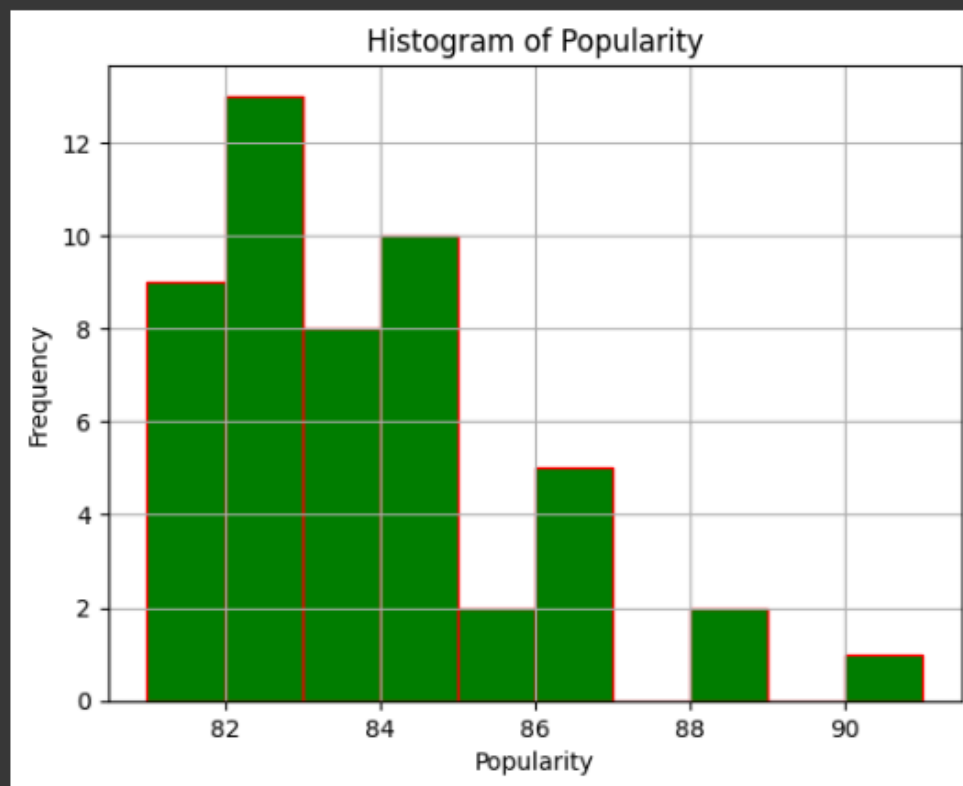
3. Histogram of Energy:

```python
import matplotlib.pyplot as plt

data = song.head(50)

# 3. Histogram

plt.hist(data['popularity'], color = 'green', edgecolor = 'red')
plt.title("Histogram of Popularity")
plt.xlabel("Popularity")
plt.ylabel("Frequency")
plt.grid()
plt.show()
```

~ Introduction:

The dataset includes information about various songs, such as their title, artist, top genre, year, beats per minute, energy, danceability, loudness, liveness, valence, length, acousticness, speechiness, and popularity. Each of these attributes helps describe different aspects of the songs. In this analysis, by visualizing this data I aim to understand how these factors may influence a song's success in the music industry or to be part of the top songs.

~ Questions:

1.   How does a song's danceability relate to its popularity?

1. Scatter Plot of Danceability vs. Popularity -

The scatter plot shows a moderate positive relationship between danceability and popularity, indicating that songs with higher danceability are usually more popular to people or an individual.

2.   Are there any notable or known trends in the energy levels of the songs over the years?

2. Line Plot of Popularity Over Index -

The line plot reveals variations in popularity among the songs, with noticeable peaks and valleys, suggesting that while some songs are very successful, others may not connect as well with listeners because they have different perspectives and as well as their preferences.

3.   What is the distribution of the top genres among the first 50 songs?

3. Histogram of Energy -

The histogram demonstrates that most songs have moderate to high energy levels, with fewer songs in the low-energy category. This may imply a preference for more energetic tracks in popular music that sometimes people tend to use these as music to a dance.

~ Insight

My insight is that songs with higher danceability tend to be more popular, which suggests that listeners prefer music that is easier to dance to. The variations in popularity further show that while certain songs shine, popularity can jump and depends on the trend, preferred, or even preference across the dataset. Additionally, the higher energy levels in the songs shows that it might reflect current trends in the music industry, where lively and upbeat tracks are more

likely to attract listeners. This insight can guide artists and producers regarding the factors that may contribute to a song's success

6. Assessment Rubrics (Grade)