Data Visualization Using Python

Data visualization is the presentation of data in a graphical or visual format. It involves using visual elements such as charts, graphs, and maps to represent and communicate complex information, patterns, and insights from data. The primary goal of data visualization is to make data more accessible, understandable, and interpretable for both technical and non-technical audiences.

Key aspects of data visualization include:

Communication: Data visualization serves as a powerful tool for conveying information clearly and concisely. It allows for the effective communication of trends, patterns, and relationships within data.

Exploration and Analysis: Visualization enables data analysts, scientists, and decision-makers to explore and analyze large datasets more efficiently. Visual representations can reveal hidden patterns or outliers that might not be immediately apparent in raw data.

Decision-Making: Well-designed visualizations facilitate informed decision-making by providing a visual summary of complex information. Decision-makers can quickly grasp insights and trends, leading to more effective and data-driven decisions.

Storytelling: Data visualization is often used to tell a story or present a narrative. By structuring data in a visually compelling way, it becomes easier to engage and persuade the audience.

Common types of data visualizations include:

Bar charts and histograms: Used to represent categorical data or the distribution of numerical data.

Line charts: Suitable for displaying trends over time or relationships between variables.

Scatter plots: Show the relationship between two continuous variables, highlighting patterns or correlations.

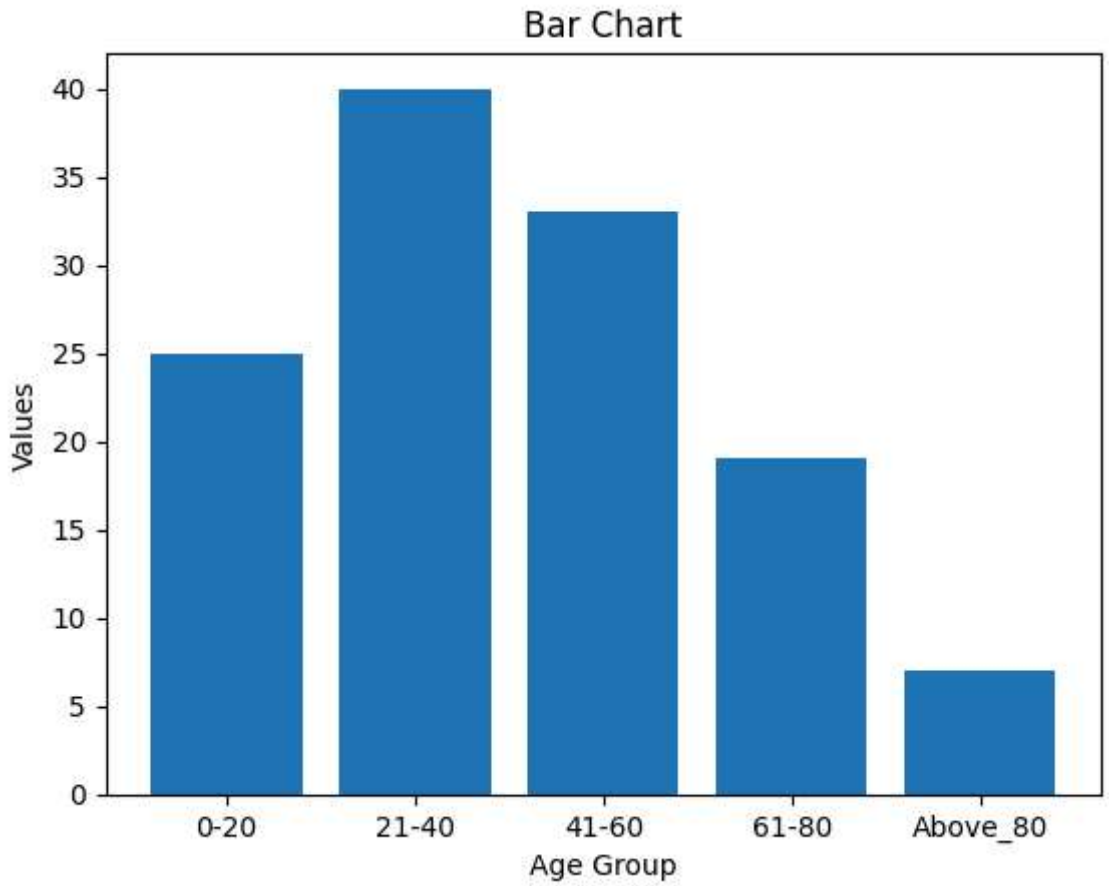
Pie charts: Represent parts of a whole, illustrating the proportion of different categories.

Heatmaps: Visualize the intensity of values in a matrix using colors.

Maps: Display geographical data and spatial relationships.

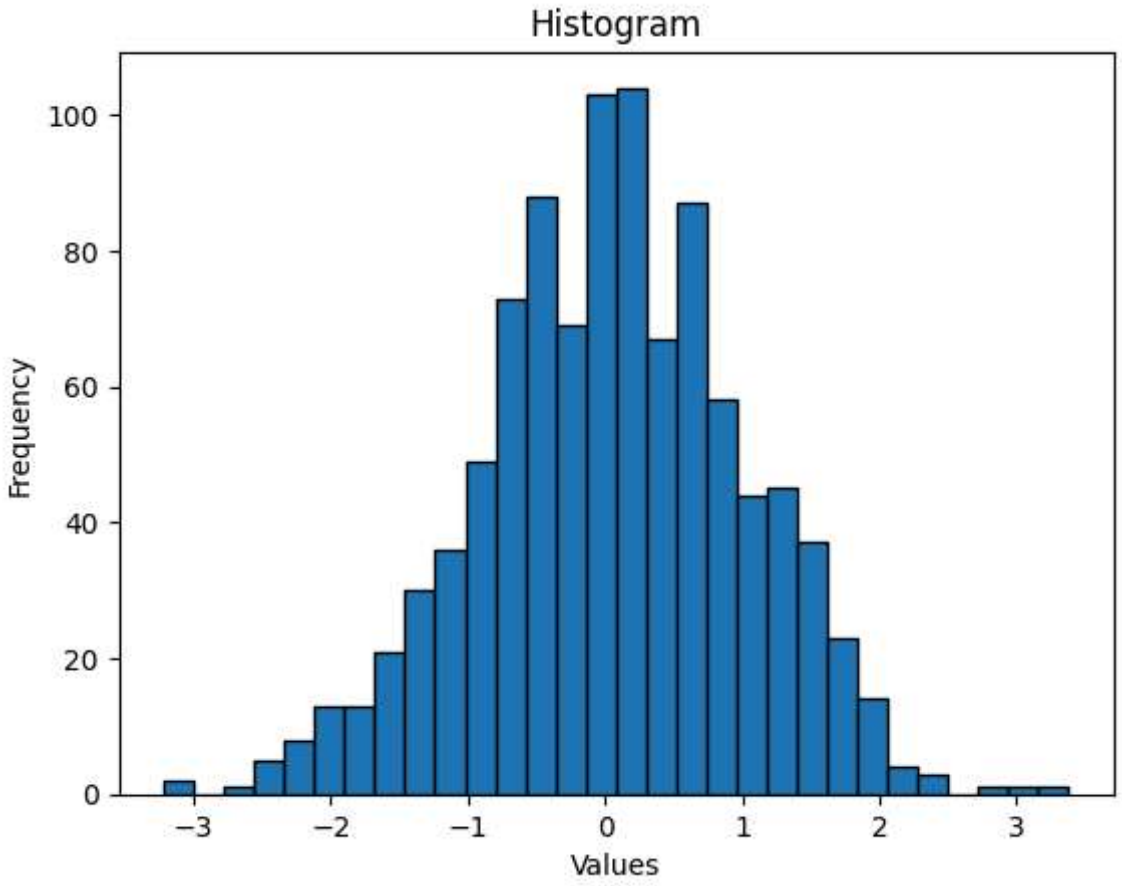
Popular tools for creating data visualizations include Matplotlib, Seaborn, Plotly, Tableau, and Microsoft Power BI, among others. Effective data visualization is not only about choosing the right charts but also involves thoughtful design choices to enhance clarity, accuracy, and the overall user experience.

```
##Importing libraries

import pandas as pd
 #for data manipulation and analysis. Useful for cleaning, filtering, and transforming datas
import numpy as np
# for scientific computing in Python. It provides support for large, multi-dimensional array
import matplotlib.pyplot as plt
#2D plotting library that produces static, animated, and interactive visualizations in Pythc
import seaborn as sns



##Bar Chat

age_group = ['0-20','21-40','41-60','61-80', 'Above_80' ]
values = [25, 40, 33, 19, 7]

plt.bar(age_group, values)
plt.title('Bar Chart')
plt.xlabel('Age Group')
plt.ylabel('Values')
plt.show()
```



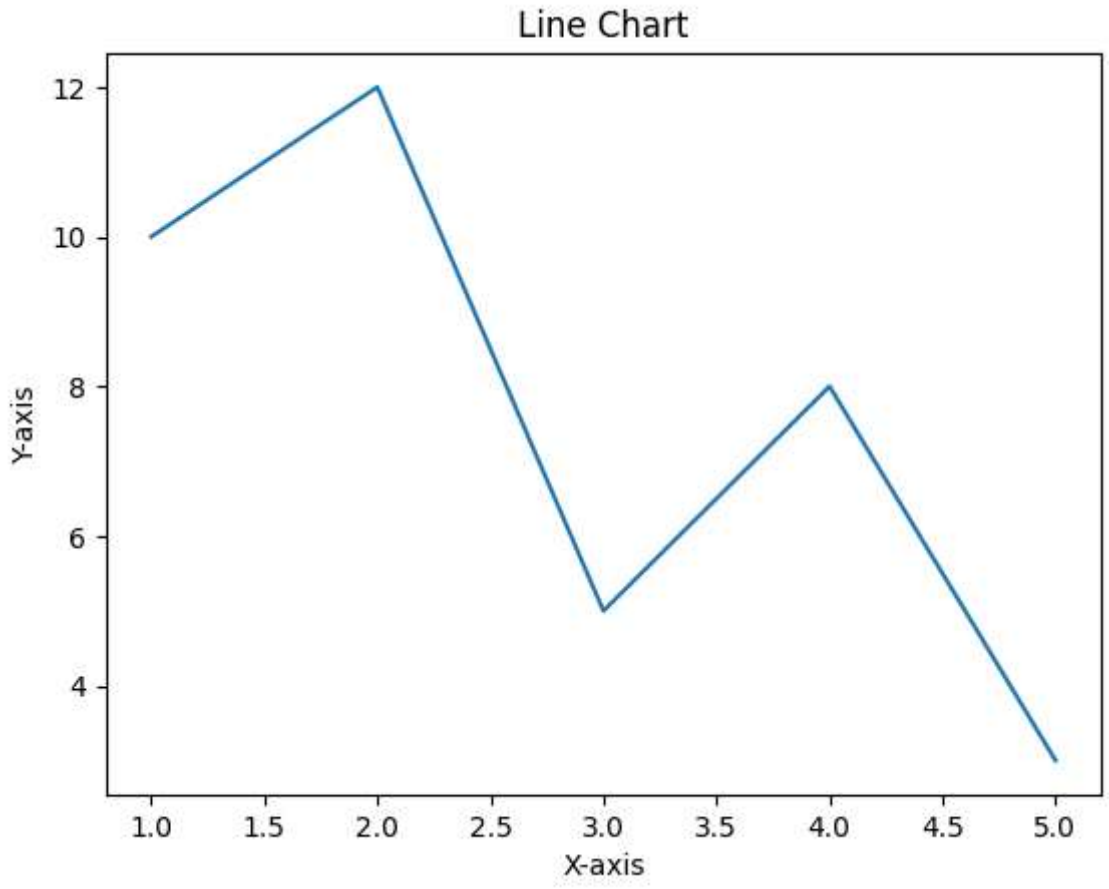
Histogram

```
data = np.random.randn(1000)

plt.hist(data, bins = 30, edgecolor='black')
plt.title('Histogram')
plt.xlabel('Values')
plt.ylabel('Frequency')
plt.show()
```

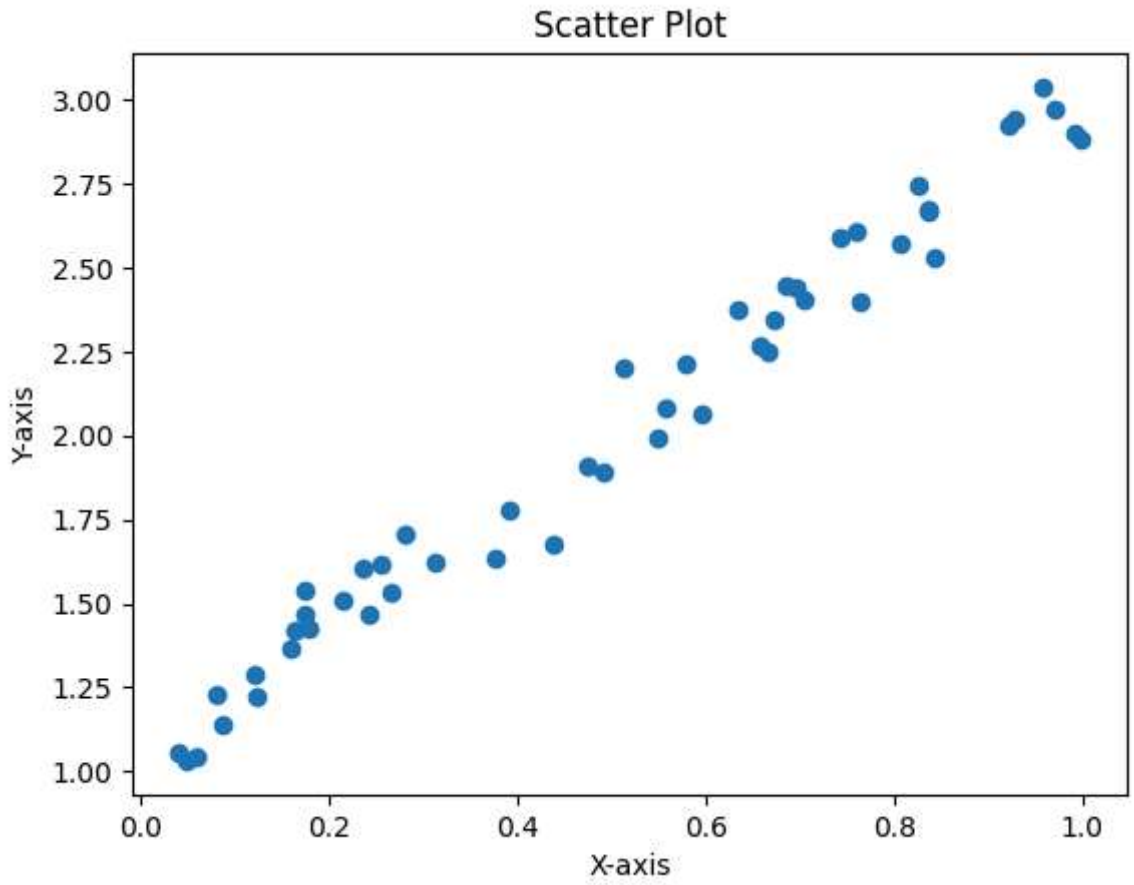


Line Chat

```
x = [1, 2, 3, 4, 5]
y = [10, 12, 5, 8, 3]


plt.plot(x,y)
plt.title('Line Chart')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.show()
```




```
x = np.random.rand(50)
y = 2 * x + 1 +0.1 * np.random.randn(50)


plt.scatter(x,y)
plt.title('Scatter Plot')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.show()
```

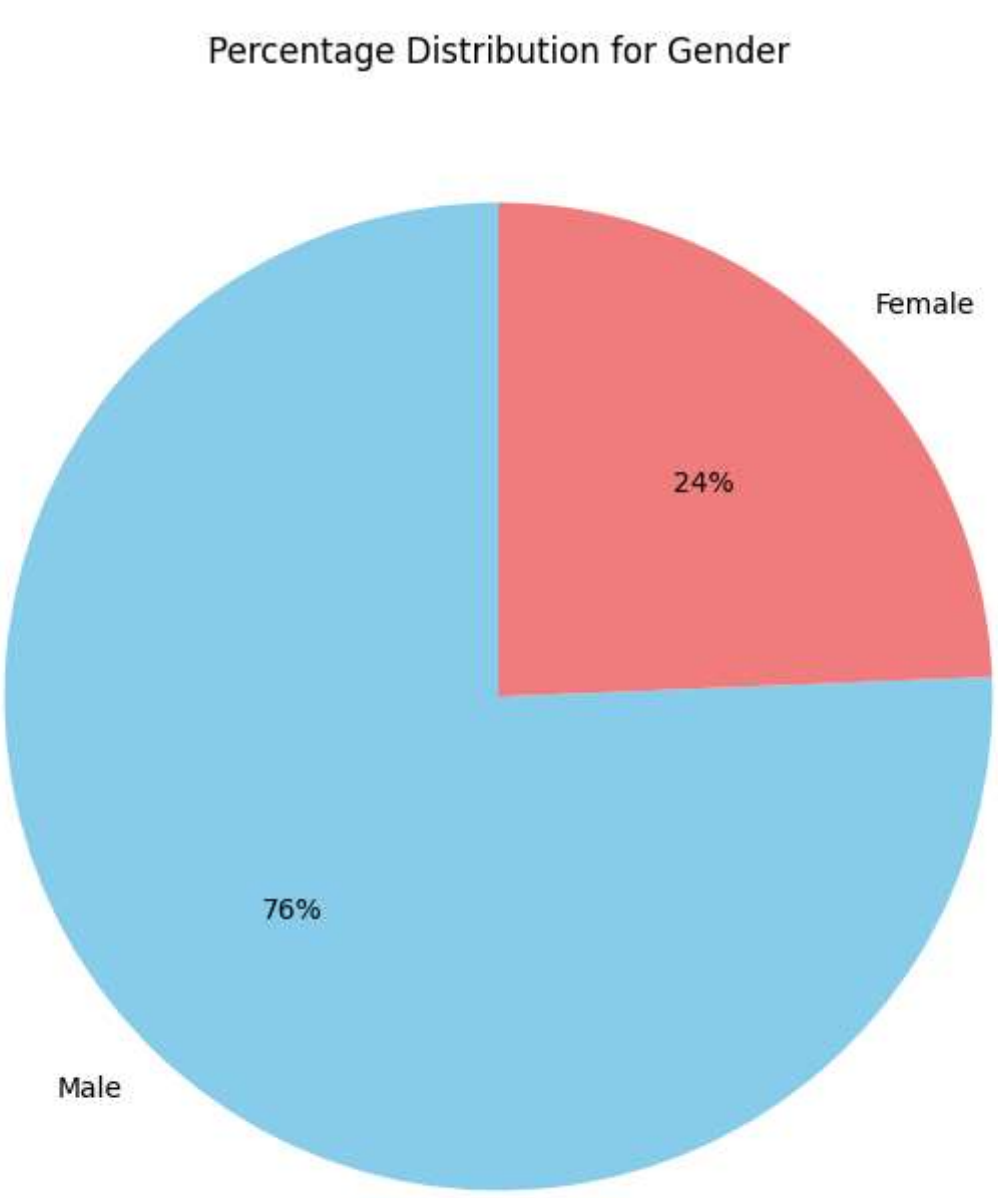



```
# Pie chat
# loading and reading dataset
ds_path = r"/content/cleaned_data.csv"
df = pd.read_csv(ds_path)
print(df.head)
```

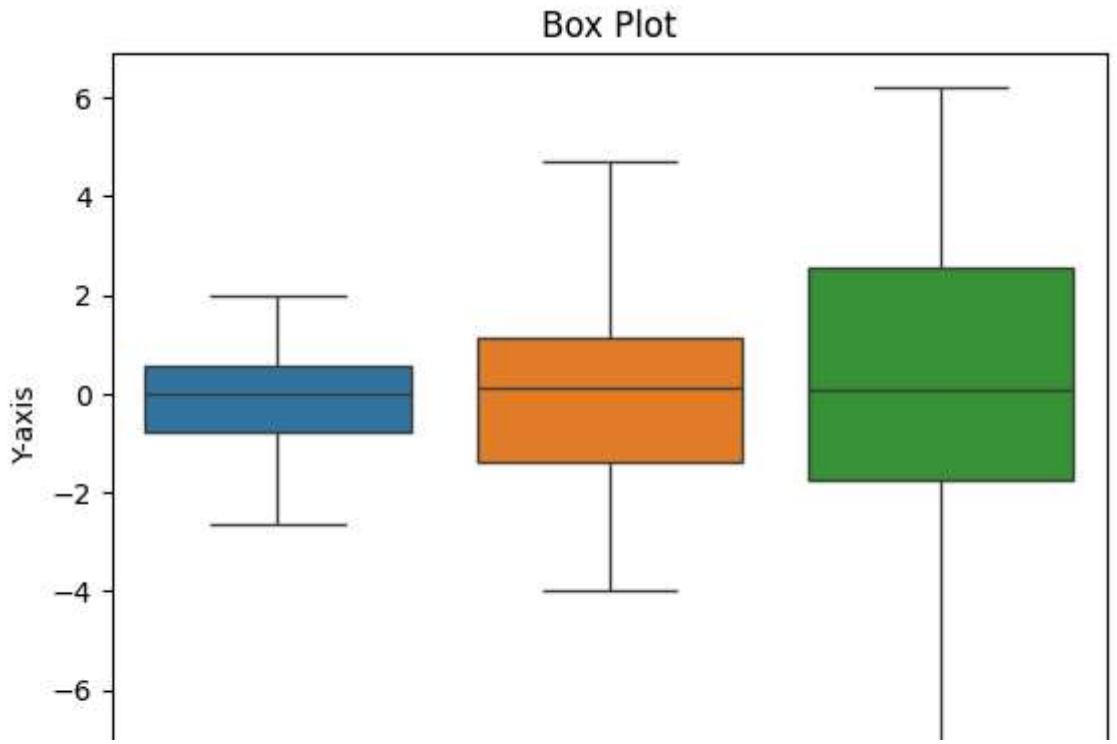
```
<bound method NDFrame.head of       ...1       start         end _submission_time FA_C
0        1  2023-06-29  2023-06-29       2023-06-30     ANE    12737    1735
1        2  2023-09-13  2023-09-13       2023-09-14     GYH    14487    7027
2        3  2023-09-28  2023-09-28       2023-09-28     BGO    12027    9019
3        4  2023-07-27  2023-07-27       2023-07-27     MGA     5802    3667
4        5  2023-08-14  2023-08-14       2023-08-21     MLF     7449    5101
...    ...         ...         ...              ...     ...      ...     ...
1809  1810  2023-07-04  2023-07-05       2023-07-05     NAS    15401    2209
1810  1811  2023-08-11  2023-08-11       2023-08-11     BGO     7240    4246
1811  1812  2023-06-26  2023-06-26       2023-06-30     NFL    16808    1718
1812  1813  2023-09-18  2023-09-18       2023-09-18     NPR     3517    7336
1813  1814  2023-08-22  2023-08-22       2023-08-22     KAR    15133    5371

        locationid       villagenam   residence  ... Thinkpeoplewerewatchingy  \
0     I41607189001     NABIDONGHA C  Peri_urban  ...                     Never
1     I31501165003      NAKIGO II A       Rural  ...                    Rarely
2     M20901008002         MBALE TC       Rural  ...                 Sometimes
3     I31201305001         BUSEYI B       Rural  ...                     Never
4     I10503309001  BUSOWOBI CENTRA       Rural  ...                     Never
```

```
...           ...              ...         ...  ...                    ...
1809  I21002157001       NAWANSINGE       Rural  ...                  Never
1810  I10503125001  BUSOWOBI CENTRA       Rural  ...                  Never
1811  M20704062004            WANTE       Rural  ...                  Never
1812  I31401882003  BULUBANDI - CEN  Peri_urban  ...                  Never
1813  I10404015001       NAWAMPENDO       Rural  ...                 Rarely

     Thinkpeoplewereagainstyo  Havemoodswings Feelshorttempered  \
0                       Never           Never             Never
1                   Sometimes       Sometimes             Never
2                   Sometimes           Never             Never
3                       Never           Never            Rarely
4                       Never           Never             Never
...                       ...             ...               ...
1809                    Never           Never             Never
1810                    Never           Never             Never
1811                    Never           Never             Never
1812                    Never           Never             Never
1813                   Rarely          Rarely            Rarely

      Thinkabouthurtingyourself  Didyouhaveanurgetodrin  \
0                         Never                   Never
1                         Never                   Never
2                         Never                   Never
3                         Never                   Never
4                         Never                   Never
...                         ...                     ...
1809                      Never                   Never
1810                      Never                   Never
1811                      Never                   Never
1812                      Never                   Never
1813                      Never               Sometimes

      Didanyonetalktoyouabout  Didyoutrytohideyourdri  \
0                         NaN                     NaN
1                         NaN                     NaN
2                         NaN                     NaN
3                         NaN                     NaN
```

```
## Pie charts

#Gender distribution
category_counts = df['gender'].value_counts()

# Calculate percentages and round to 0 decimal places
category_percentages = (category_counts / len(df['gender'])) * 100

plt.figure(figsize=(8, 8))
plt.pie(category_percentages, labels=['Male', 'Female'], autopct='%1.0f%%', startangle=90, c
plt.title('Percentage Distribution for Gender')
plt.show()
```




```
## Box Plot

data = [np.random.normal(0, std, 100) for std in range(1, 4)]
sns.boxplot(data=data)
plt.title('Box Plot')
plt.xlabel('X-axis')
plt.ylabel('Y-axis')
plt.show()
```




```
## HEatmap

data = np.random.rand(10,10)

sns.heatmap(data, annot=True)
plt.title('Heatmap')
plt.show()
```