# Programming with Python

## Visualization

# Agenda

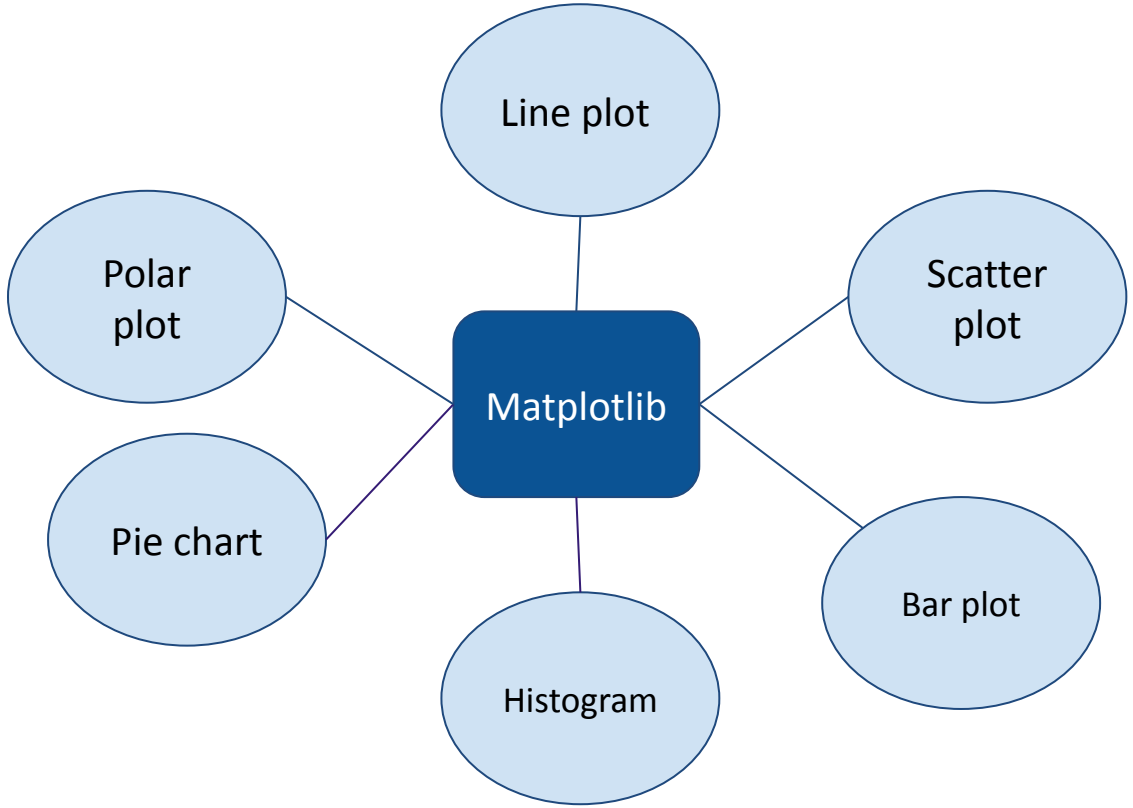
- Visualization in Python
- Matplotlib
- Pyplot
- Different plots in Matplotlib
- Seaborn
- Different plots in Seaborn
- Summary

# Python Visualization

- Visual description of data.
- Provides better understanding of data.
- Python provides variety of libraries, e.g.: Matplotlib, Seaborn, ggplot, Plotly, etc.
- Helps to explore patterns, trends, and correlations.
- Helps to present reports, outcomes, and inferences in visual format.

# Matplotlib

- Python library for visualization
- Build on NumPy arrays
- “conda install -c conda-forge matplotlib”
- Pyplot class supports wide variety of plots

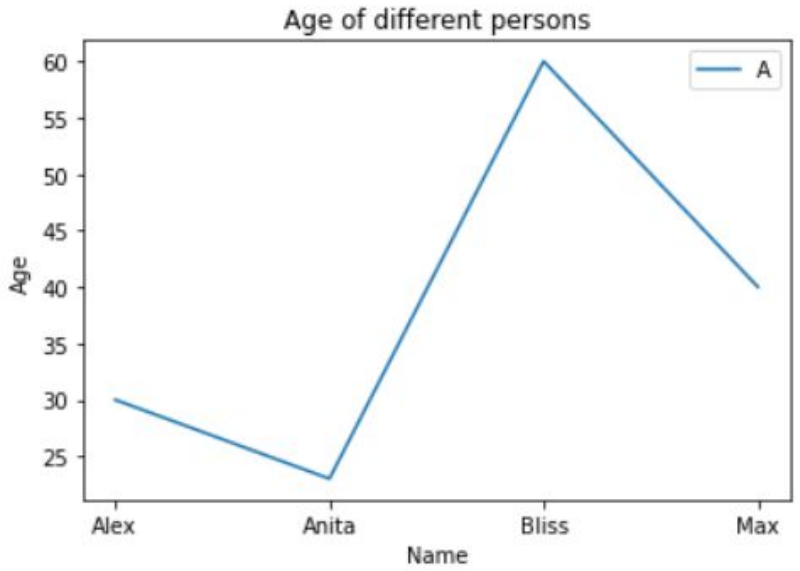


# Pyplot

- Matplotlib module that provides collection of functions for plotting
- Gives MATLAB like interface to matplotlib

**Line Plot**

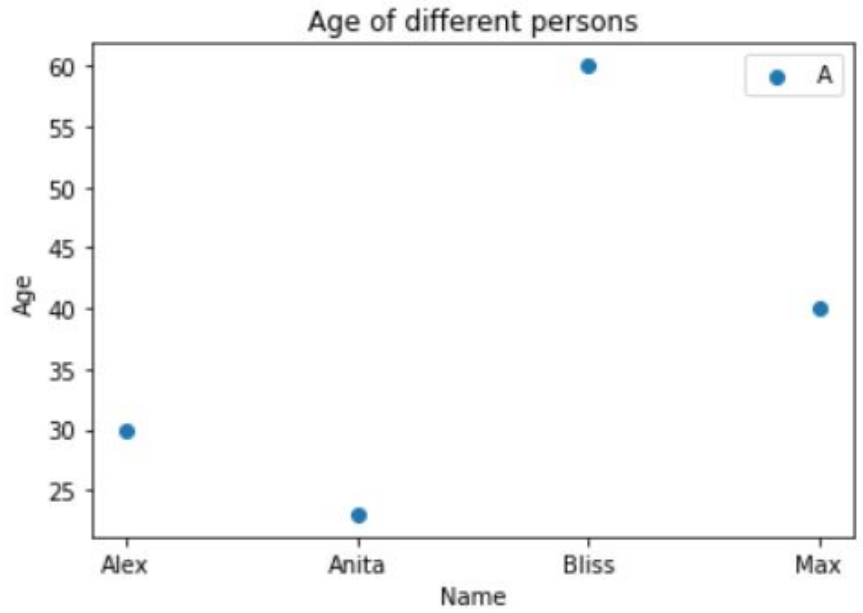
```
import pandas as pd
from matplotlib import pyplot as plt
df = pd.read_csv("Week4_Profile_Data.csv")
plt.plot(df["Name"], df["Age"])
plt.xlabel("Name")
plt.ylabel("Age")
plt.title('Age of different persons')
plt.legend("Age")
plt.show()
```



# Different plots in Matplotlib

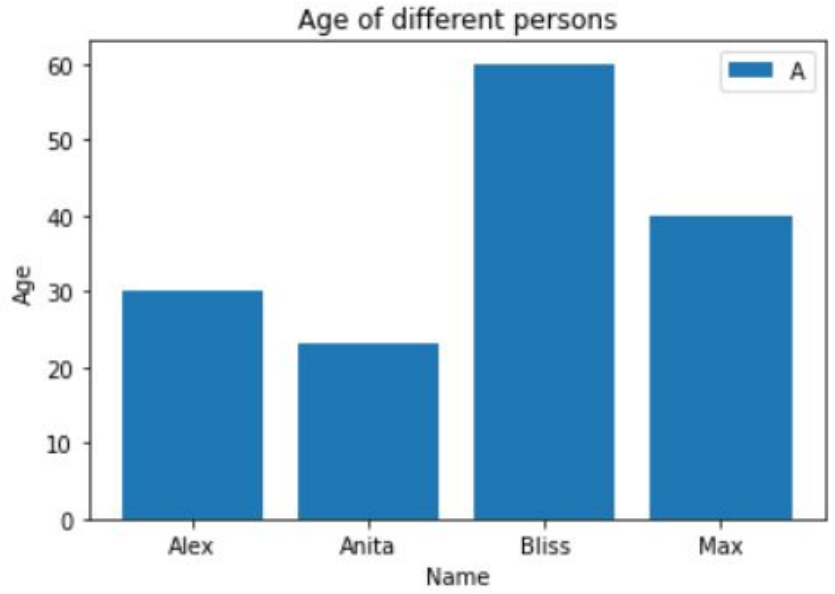
**Scatter Plot**

```
plt.scatter(df["Name"], df["Age"])
plt.xlabel("Name")
plt.ylabel("Age")
plt.title('Age of different persons')
plt.legend("Age")
plt.show()
```



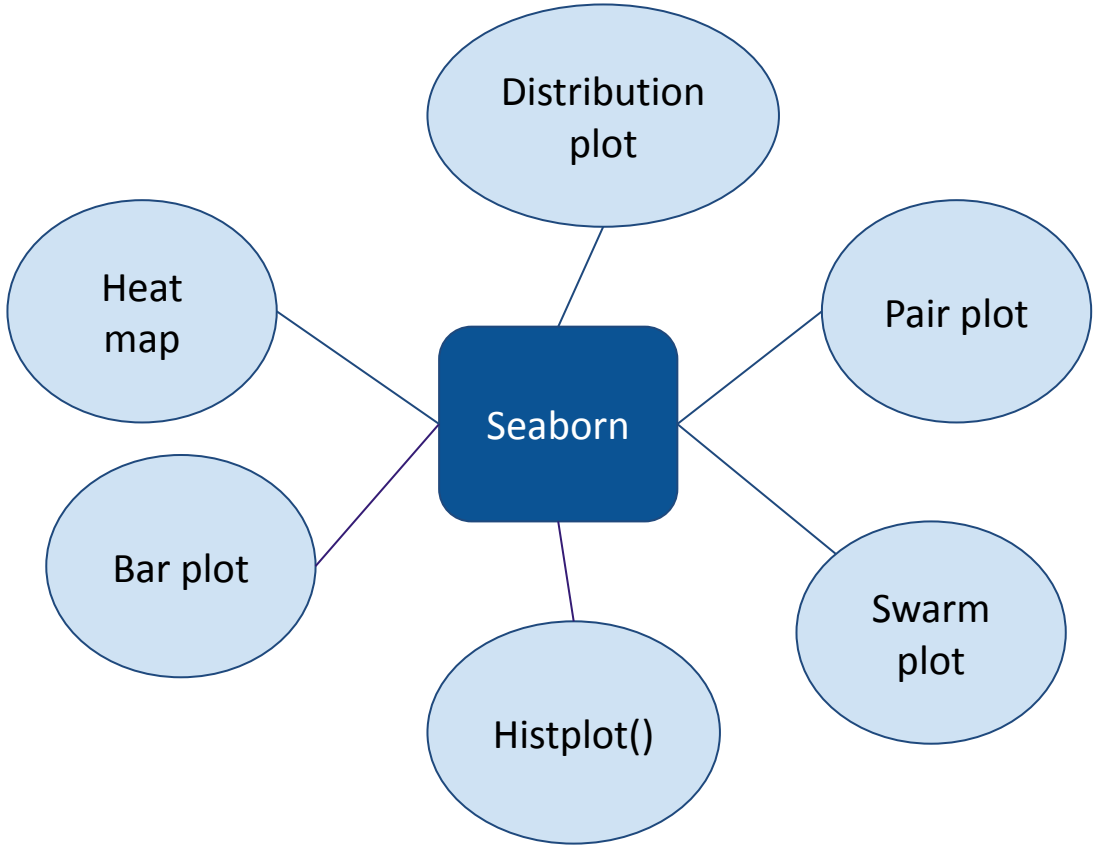

**Bar plot**

```
plt.bar(df["Name"], df["Age"])
plt.xlabel("Name")
plt.ylabel("Age")
plt.title('Age of different persons')
plt.legend("Age")
plt.show()
```



# Seaborn

- Build on top of Matplotlib
- Integrated with Pandas data structure
- Command : conda install -c anaconda seaborn
- Makes statistical plots more attractive



# Different plots in Seaborn

- Distribution plot for continuous data
  - displot()
  - histplot()
  - kdeplot()
- Categorical plots for categorical data
  - barplot()
  - countplot()
  - boxplot()
  - violinplot()
  - swarmplot()
  - stripplot()

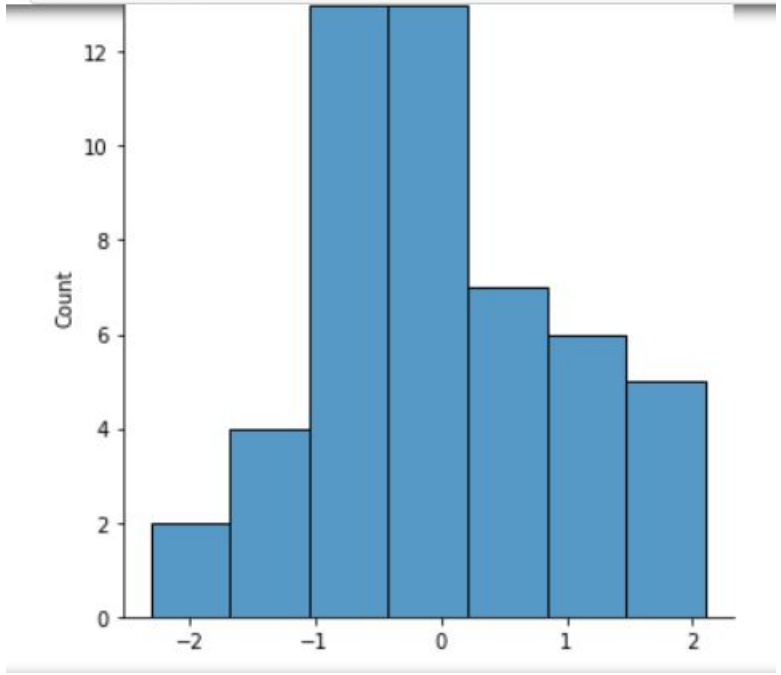
# Different plots in Seaborn

- Relational plots for checking relationship between two or more variables
  - joinplot()
  - pairplot()
  - scatterplot()
  - relplot()
  - lineplot()
  - heatmap()

# Different plots in Seaborn

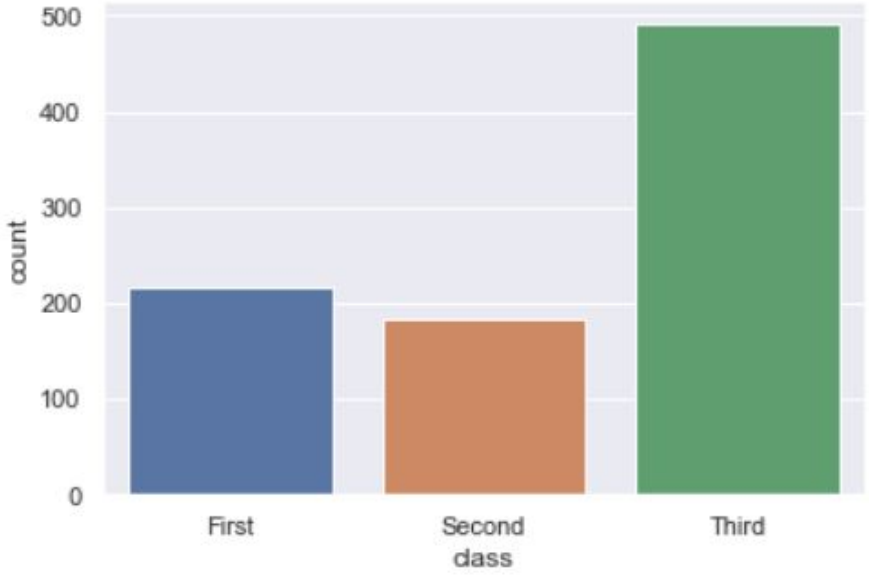
**Displot**

```
: import seaborn as sns
  import numpy as np
  np.random.seed(1)
  random_values = np.random.randn(50)
  ax_grid = sns.displot(random_values)
  plt.show()
```




**Count plot**

```
sns.set_theme(style="darkgrid")
titanic_values = sns.load_dataset("titanic")
ax_grid = sns.countplot(x="class", data=titanic_values)
```

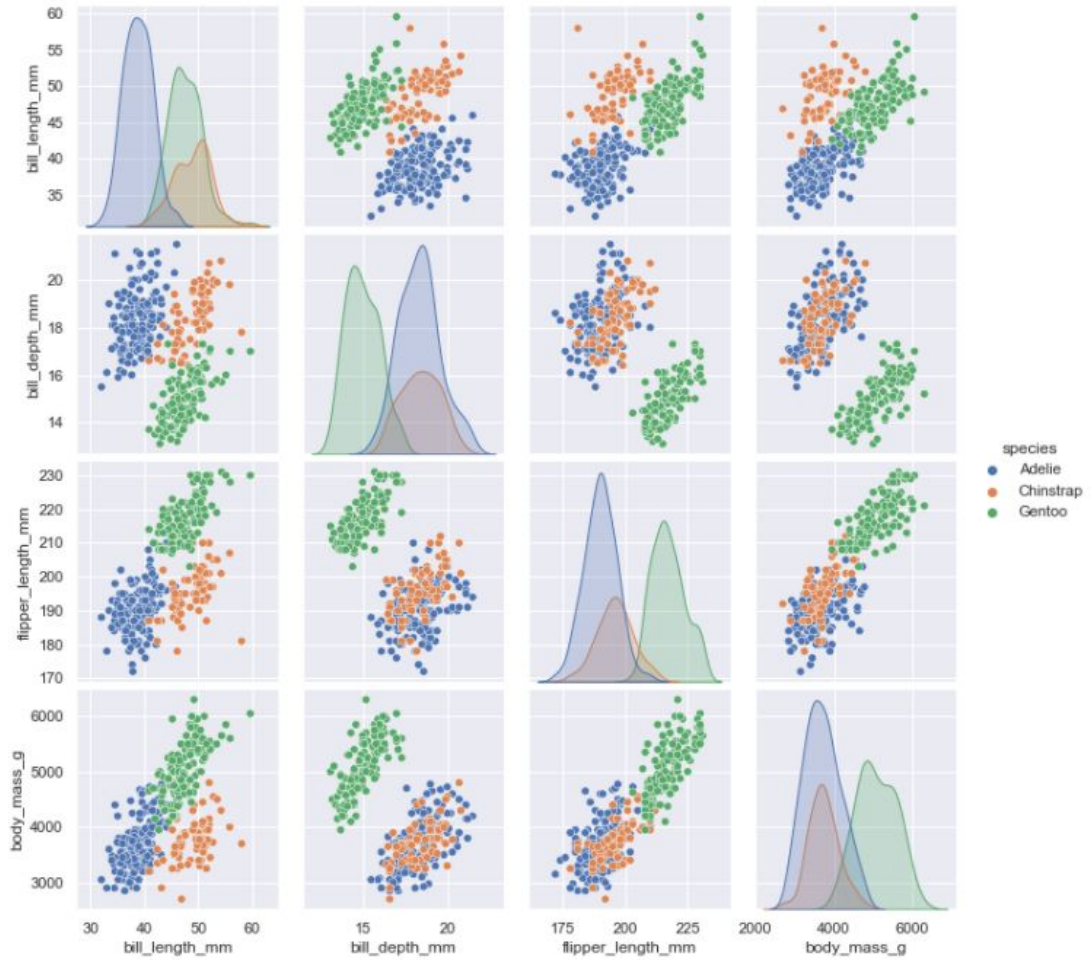


# Pair plot

Pair plot

```
penguins_data = sns.load_dataset("penguins")
sns.pairplot(penguins_data, hue="species")
```

```
<seaborn.axisgrid.PairGrid at 0x1e93d9e6fa0>
```



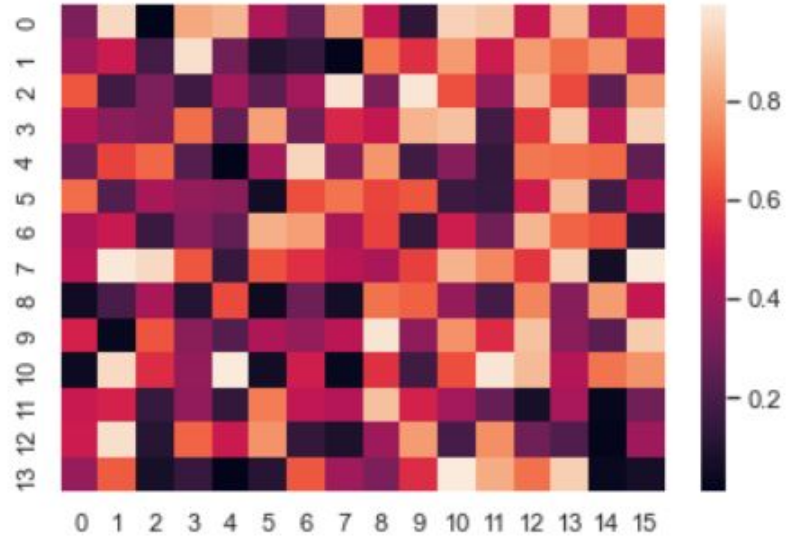

- Creates an axis grid
- Each variable is represented as row on y-axis and as column on x-axis
- Pairwise relationship is represented through scatter plot
- Distribution is shown through displot

# Heat map in Seaborn

- Values are represented in terms of color encoded matrix
- Color maps (cmap=) can have different palette
- Values are directly proportional to color of each matrix
- i.e. low value has low intensity color

**Heat map**

```
sns.set_theme()
uniform_values = np.random.rand(14, 16)
ax_grid = sns.heatmap(uniform_values)
plt.show()
```



# Summary

- Discussed visualization techniques in Python.
- Matplotlib and use of Pyplot module.
- Different plots associated with Matplotlib.
- Seaborn in Python
- Different plots associated with Seaborn.

# Hands on

# THANK YOU

# Happy learning ☺️