Experiment 11

Aim: To study Pandas / Matplotlib in Python

Theory:

Using pandas:

- Show various operations using dataframe to read data , clean data and analyse data.
- Create series, create own dataframe
- Readcsv
- Delete NA values from the dataframe(all NA and NA values of specific columns)
- Fill NA values with random values, mean, median)
- display statistical information of the data frame
- Establish relationship between the columns of the data frame

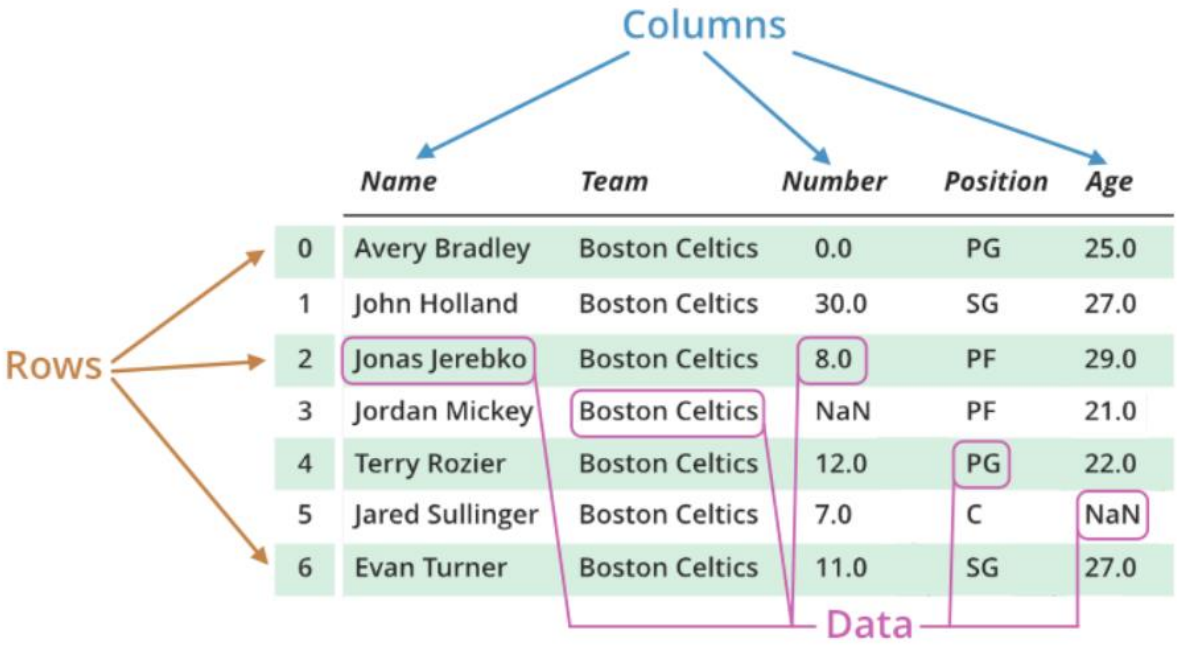
using matplotlib

- plot follwoing graphs
- Barchart
- piechart
- Scatter plot
- Histogram

PANDAS:

Pandas is a software library written for the Python programming language for data manipulation and analysis. It offers data structures and operations for manipulating numerical tables and time series. Pandas is mainly used for data analysis. Pandas allows importing data from various file formats such as comma-separated values, JSON, SQL database tables or queries, and Microsoft Excel. Pandas allows various data manipulation operations such as merging, reshaping, selecting, as well as data cleaning, and data wrangling features which include DataFrame object for data manipulation with integrated indexing, data alignment and integrated handling of missing data, reshaping and pivoting of data sets, label-based slicing and subsetting of large data sets, data set merging and joining, Hierarchical axis indexing, Time series-functionality moving window statistics, moving window linear regressions, date shifting and lagging and Data filtration. The library is highly optimized for performance, with critical code paths written in Cython or C.

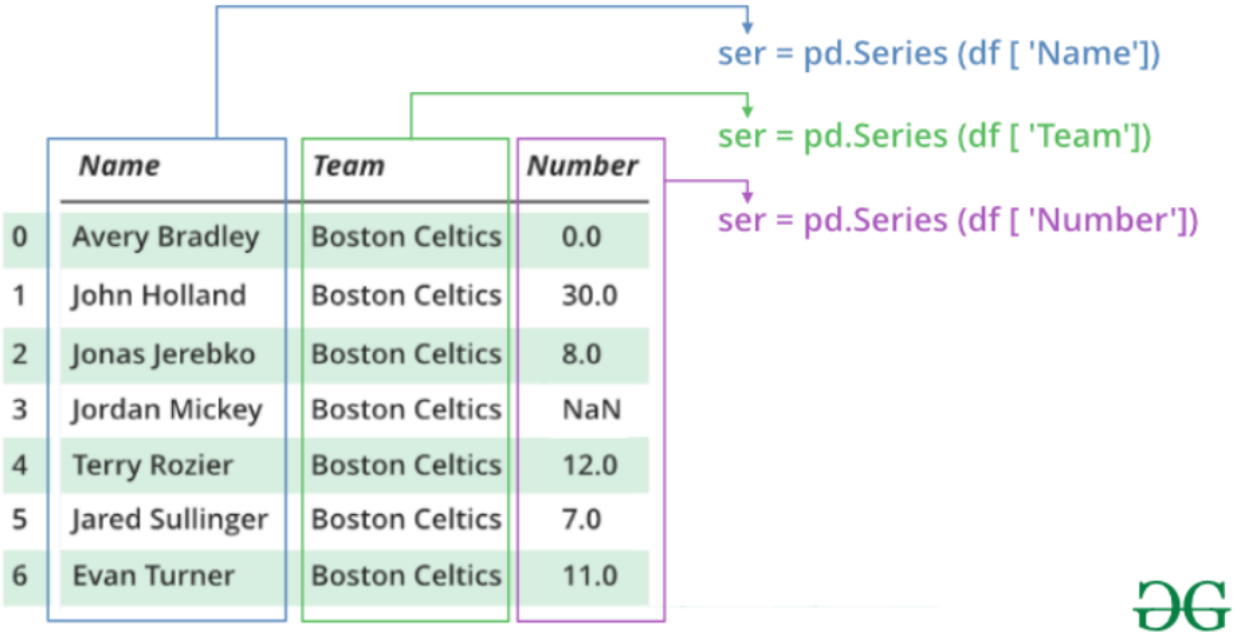
Pandas DataFrame is two-dimensional size-mutable, potentially heterogeneous tabular data structure with labeled axes (rows and columns). A Data frame is a two-dimensional data structure, i.e., data is aligned in a tabular fashion in rows and columns. Pandas DataFrame consists of three principal components, the **data**, **rows**, and **columns**.



	Name	Team	Number	Position	Age
0	Avery Bradley	Boston Celtics	0.0	PG	25.0
1	John Holland	Boston Celtics	30.0	SG	27.0
2	Jonas Jerebko	Boston Celtics	8.0	PF	29.0
3	Jordan Mickey	Boston Celtics	NaN	PF	21.0
4	Terry Rozier	Boston Celtics	12.0	PG	22.0
5	Jared Sullinger	Boston Celtics	7.0	C	NaN
6	Evan Turner	Boston Celtics	11.0	SG	27.0

Series

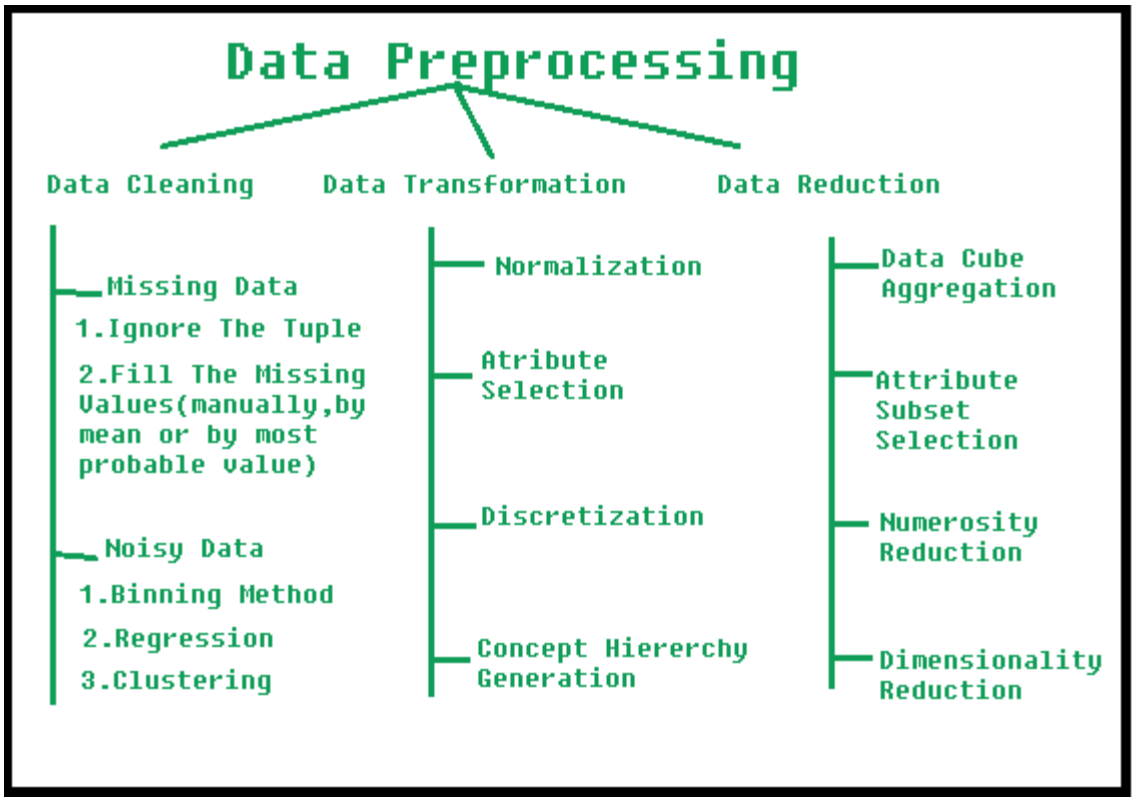
Pandas Series is a one-dimensional labeled array capable of holding data of any type (integer, string, float, python objects, etc.). The axis labels are collectively called *index*. Pandas Series is nothing but a column in an excel sheet.
Labels need not be unique but must be a hashable type. The object supports both integer and label-based indexing and provides a host of methods for performing operations involving the index.



	Name	Team	Number
0	Avery Bradley	Boston Celtics	0.0
1	John Holland	Boston Celtics	30.0
2	Jonas Jerebko	Boston Celtics	8.0
3	Jordan Mickey	Boston Celtics	NaN
4	Terry Rozier	Boston Celtics	12.0
5	Jared Sullinger	Boston Celtics	7.0
6	Evan Turner	Boston Celtics	11.0

Here are the basic data cleaning tasks we tackle:

- Importing Libraries
- Input Customer Feedback Dataset
- Locate Missing Data
- Check for Duplicates
- Detect Outliers
- Normalize Casing



MATPLOTLIB:

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK. There is also a procedural "pylab" interface based on a state machine (like OpenGL), designed to closely resemble that of MATLAB, though its use is discouraged. SciPy makes use of Matplotlib. Several toolkits are available which extend Matplotlib functionality. 1. Basemap: map plotting with various projections, coastlines, and political boundaries 2. Cartopy: a mapping library featuring object-oriented map projection definitions, and arbitrary point, line, polygon, and image transformation capabilities 3. Excel tools: utilities for exchanging data with Microsoft Excel 4. GTK tools: interface to the GTK library 5. Mplot3d: 3-D plots 6. Natgrid: interface to the natgrid library for gridding irregularly spaced data. 7. Seaborn: provides an API on top of Matplotlib that offers sane choices for plot style and color defaults and defines simple high-level functions.

Code & Output:

PANDAS:

```
import pandas as pd
player=['A','B','C','D','E','F','G','H','I','J']
match=[64,54,62,None,59,35,22,15]
player_series=pd.Series(player)
match_series=pd.Series(match)
frame={'Player':player_series,'Match':match_series}
result = pd.DataFrame(frame)
print(result)
```

```
  Player  Match
0      A   64.0
1      B   54.0
2      C   62.0
3      D    NaN
4      E   59.0
5      F   35.0
6      G   22.0
7      H   15.0
8      I    NaN
9      J    NaN
```

```
[5] df=pd.read_csv('/content/titanic.csv')
    df.head()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	892	0	3	Kelly, Mr. James	male	34.5	0	0	330911	7.8292	NaN	Q
1	893	1	3	Wilkes, Mrs. James (Ellen Needs)	female	47.0	1	0	363272	7.0000	NaN	S
2	894	0	2	Myles, Mr. Thomas Francis	male	62.0	0	0	240276	9.6875	NaN	Q
3	895	0	3	Wirz, Mr. Albert	male	27.0	0	0	315154	8.6625	NaN	S
4	896	1	3	Hirvonen, Mrs. Alexander (Helga E Lindqvist)	female	22.0	1	1	3101298	12.2875	NaN	S

```
df.isnull()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	False	False	False	False	False	False	False	False	False	False	True	False
1	False	False	False	False	False	False	False	False	False	False	True	False
2	False	False	False	False	False	False	False	False	False	False	True	False
3	False	False	False	False	False	False	False	False	False	False	True	False
4	False	False	False	False	False	False	False	False	False	False	True	False
...	...	...	...	...	...	...	...	...	...	...	...	...
413	False	False	False	False	False	True	False	False	False	False	True	False
414	False	False	False	False	False	False	False	False	False	False	False	False
415	False	False	False	False	False	False	False	False	False	False	True	False
416	False	False	False	False	False	True	False	False	False	False	True	False
417	False	False	False	False	False	True	False	False	False	False	True	False

418 rows × 12 columns

```
df.dropna()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
12	904	1	1	Snyder, Mrs. John Pillsbury (Nelle Stevenson)	female	23.0	1	0	21228	82.2667	B45	S
14	906	1	1	Chaffee, Mrs. Herbert Fuller (Carrie Constance...	female	47.0	1	0	W.E.P. 5734	61.1750	E31	S
24	916	1	1	Ryerson, Mrs. Arthur Larned (Emily Maria Borie)	female	48.0	1	3	PC 17608	262.3750	B57 B59 B63 B66	C
26	918	1	1	Ostby, Miss. Helene Ragnhild	female	22.0	0	1	113509	61.9792	B36	C
28	920	0	1	Brady, Mr. John Bertram	male	41.0	0	0	113054	30.5000	A21	S
...	...	...	...	...	...	...	...	...	...	...	...	...
404	1296	0	1	Frauenthal, Mr. Isaac Gerald	male	43.0	1	0	17765	27.7208	D40	C
405	1297	0	2	Nourney, Mr. Alfred (Baron von Drachstedt")"	male	20.0	0	0	SC/PARIS 2166	13.8625	D38	C
407	1299	0	1	Widener, Mr. George Dunton	male	50.0	1	1	113503	211.5000	C80	C

```
mean_value=df['Age'].mean()
print(mean_value)
median_value=df['Age'].median()
print(median_value)
```

```
30.272590361445783
27.0
```

```
df['Age'].fillna(value=median_value, inplace=True)
print(df)
```

```
     PassengerId  Survived  Pclass  \
0            892         0       3
1            893         1       3
2            894         0       2
3            895         0       3
4            896         1       3
..           ...       ...     ...
413         1305         0       3
414         1306         1       1
415         1307         0       3
416         1308         0       3
417         1309         0       3

                                          Name     Sex   Age  SibSp  Parch  \
0                              Kelly, Mr. James    male  34.5      0      0
1              Wilkes, Mrs. James (Ellen Needs)  female  47.0      1      0
2                     Myles, Mr. Thomas Francis    male  62.0      0      0
3                              Wirz, Mr. Albert    male  27.0      0      0
4    Hirvonen, Mrs. Alexander (Helga E Lindqvist)  female  22.0      1      1
..                                          ...     ...   ...    ...    ...
413                         Spector, Mr. Woolf    male  27.0      0      0
414               Oliva y Ocana, Dona. Fermina  female  39.0      0      0
415               Saether, Mr. Simon Sivertsen    male  38.5      0      0
416                       Ware, Mr. Frederick    male  27.0      0      0
417                  Peter, Master. Michael J    male  27.0      1      1
```

```
df.describe()
```

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	418.000000	418.000000	418.000000	418.000000	418.000000	418.000000	417.000000
mean	1100.500000	0.363636	2.265550	29.599282	0.447368	0.392344	35.627188
std	120.810458	0.481622	0.841838	12.703770	0.896760	0.981429	55.907576
min	892.000000	0.000000	1.000000	0.170000	0.000000	0.000000	0.000000
25%	996.250000	0.000000	1.000000	23.000000	0.000000	0.000000	7.895800
50%	1100.500000	0.000000	3.000000	27.000000	0.000000	0.000000	14.454200
75%	1204.750000	1.000000	3.000000	35.750000	1.000000	0.000000	31.500000
max	1309.000000	1.000000	3.000000	76.000000	8.000000	9.000000	512.329200

```
df.corr()
```

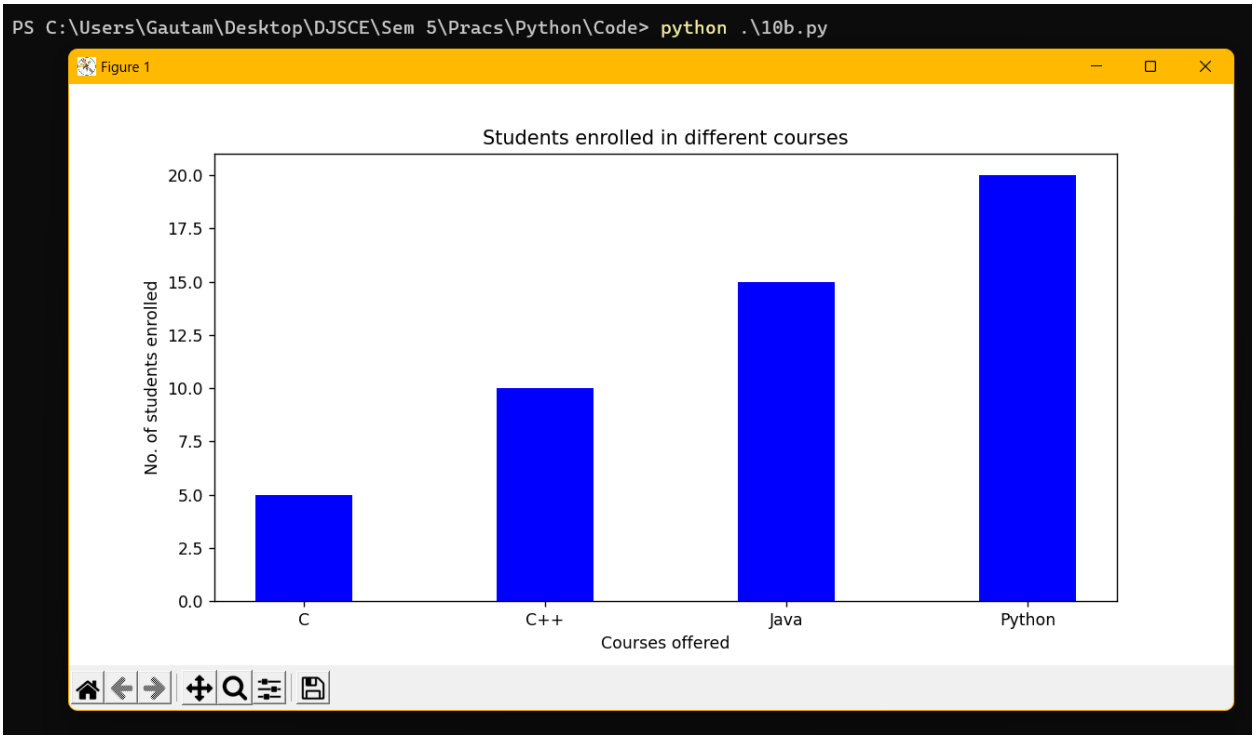
	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
PassengerId	1.000000	-0.023245	-0.026751	-0.031447	0.003818	0.043080	0.008211
Survived	-0.023245	1.000000	-0.108615	0.008035	0.099943	0.159120	0.191514
Pclass	-0.026751	-0.108615	1.000000	-0.467853	0.001087	0.018721	-0.577147
Age	-0.031447	0.008035	-0.467853	1.000000	-0.071197	-0.043731	0.347105
SibSp	0.003818	0.099943	0.001087	-0.071197	1.000000	0.306895	0.171539
Parch	0.043080	0.159120	0.018721	-0.043731	0.306895	1.000000	0.230046
Fare	0.008211	0.191514	-0.577147	0.347105	0.171539	0.230046	1.000000

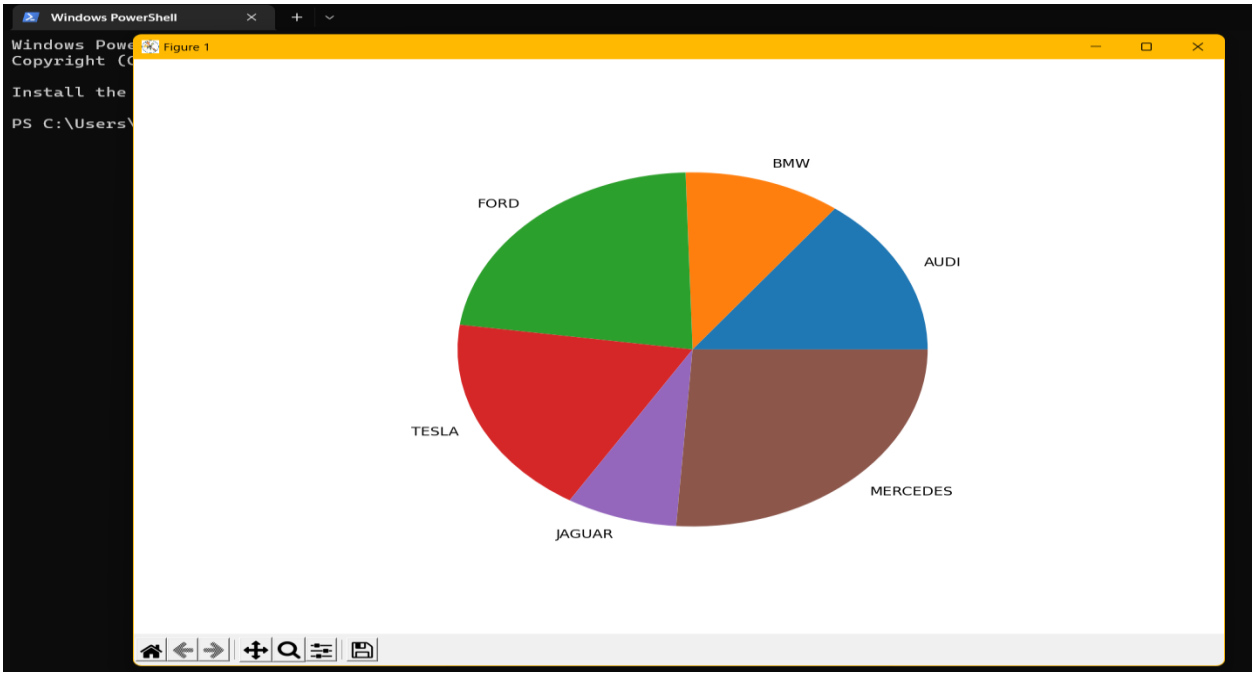
MATPLOTLIB:

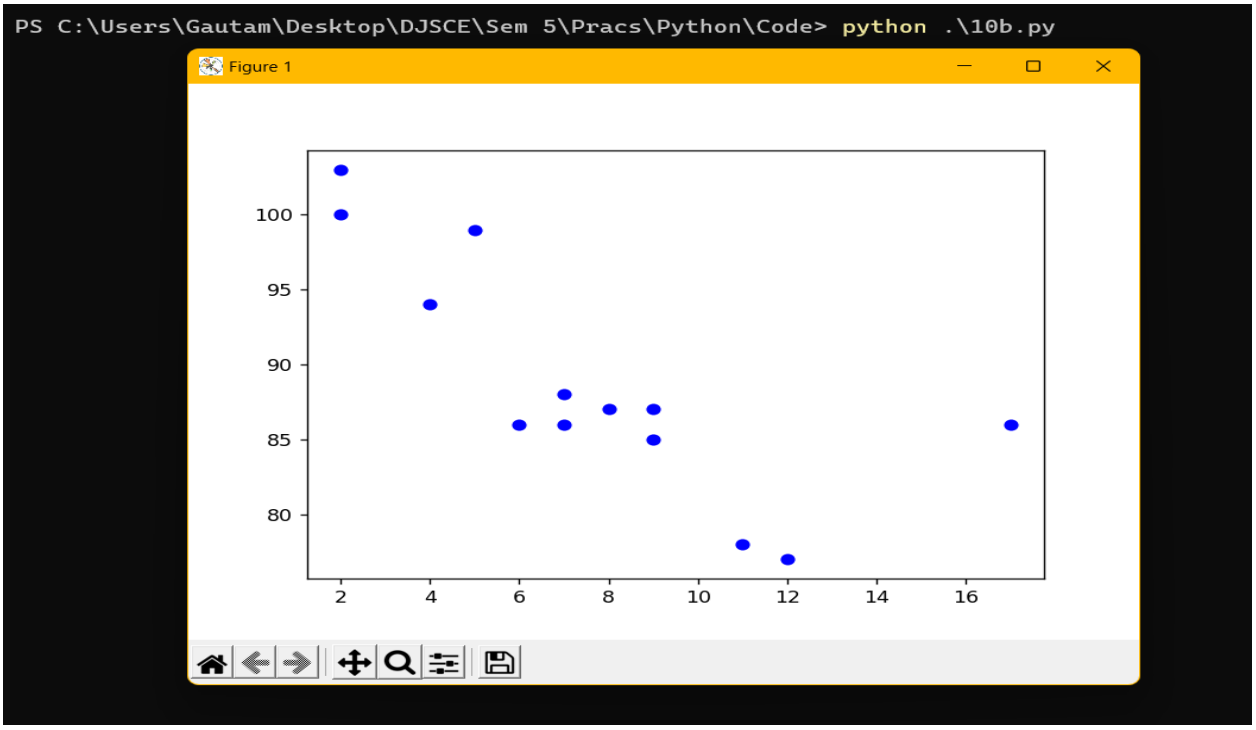
```
import numpy as np
import matplotlib.pyplot as plt

def barchart():
   data = {'C': 5, 'C++': 10, 'Java': 15,'Python': 20}
   courses = list(data.keys())
   values = list(data.values())
   fig = plt.figure(figsize=(10, 5))
   plt.bar(courses, values, color='blue',width=0.4)
   plt.xlabel("Courses offered")
   plt.ylabel("No. of students enrolled")
   plt.title("Students enrolled in different courses")
   plt.show()
```

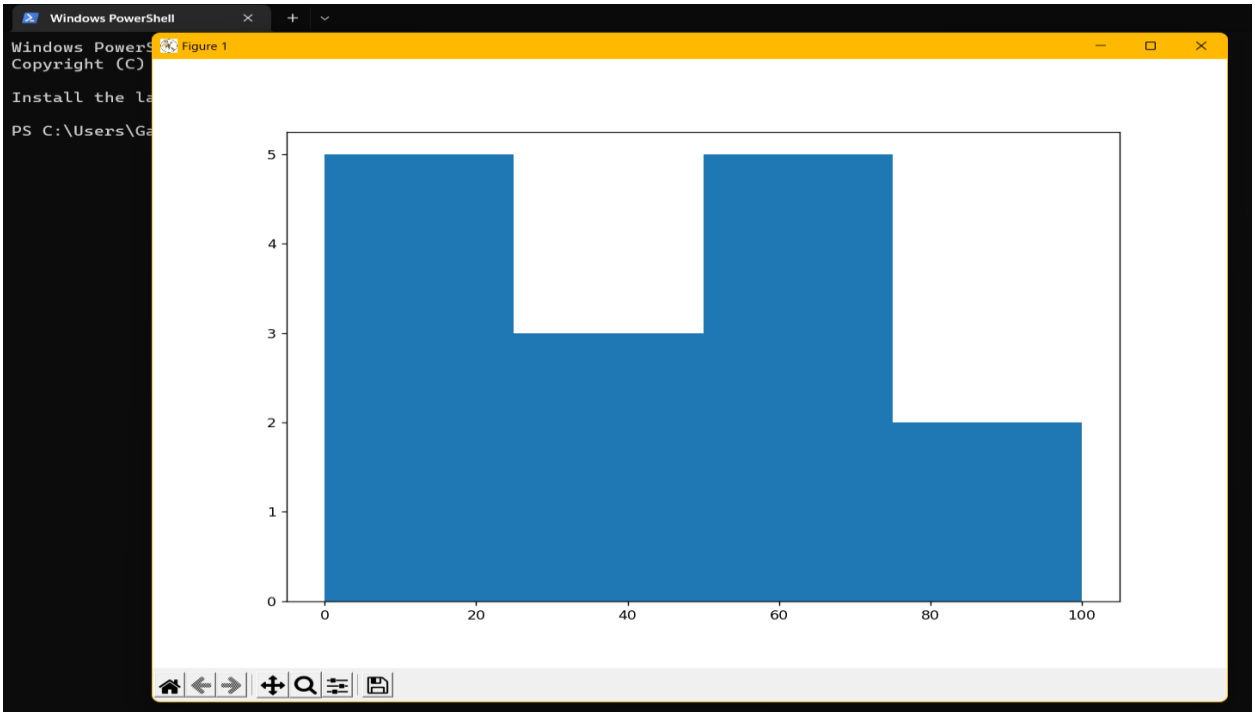
```python
def piechart():
    cars = ['AUDI', 'BMW', 'FORD','TESLA', 'JAGUAR', 'MERCEDES']
    data = [23, 17, 35, 29, 12, 41]
    fig = plt.figure(figsize=(10, 7))
    plt.pie(data, labels=cars)
    plt.show()

def scatterplot():
    x = [5, 7, 8, 7, 2, 17, 2, 9,4, 11, 12, 9, 6]
    y = [99, 86, 87, 88, 100, 86,103, 87, 94, 78, 77, 85, 86]
    plt.scatter(x, y, c="blue")
    plt.show()

def histogram():
    a = np.array([22, 87, 5, 43, 56,73, 55, 54, 11,20, 51, 5, 79, 31,27])
    fig, ax = plt.subplots(figsize=(10, 7))
    ax.hist(a, bins=[0, 25, 50, 75, 100])
    plt.show()

barchart()
piechart()
scatterplot()
histogram()
```

```
PS C:\Users\Gautam\Desktop\DJSCE\Sem 5\Pracs\Python\Code> python .\10b.py
```




Windows PowerShell
Figure 1
BMW
AUDI
FORD
TESLA
JAGUAR
MERCEDES


PS C:\Users\Gautam\Desktop\DJSCE\Sem 5\Pracs\Python\Code> python .\10b.py
Figure 1
100
95
90
85
80
2
4
6
8
10
12
14
16



Conclusion:

Thus, Pandas is used to prepare and explore data for preliminary analysis, it's used across industries and by many levels of data professionals. Thus, Matplotlib is extremely powerful because it allows users to create numerous and diverse plot types.