**LIST OF EXPERIMENTS:**

1. Working with [Numpy arrays](https://courseware.dceedu.org:6862/mod/page/view.php?id=12418)

2. Working with Pandas [data frames](https://courseware.dceedu.org:6862/mod/page/view.php?id=12519)

3. [Basic plots using Matplotlib](https://courseware.dceedu.org:6862/mod/page/view.php?id=12784)

4. [Frequency distributions](https://courseware.dceedu.org:6862/mod/page/view.php?id=13058)

5. Averages

6. Variability

7. Normal curves

8. Correlation and scatter plots

9. Correlation coefficient

10. Regression

| **Sl.N o.** | **Name of the Experiment** | **Page No.** |
| --- | --- | --- |
| 1 | Working with [Numpy arrays](https://courseware.dceedu.org:6862/mod/page/view.php?id=12418) |  |
| 2 | Working with Pandas [data frames](https://courseware.dceedu.org:6862/mod/page/view.php?id=12519) |  |
| 3 | [Basic plots using Matplotlib](https://courseware.dceedu.org:6862/mod/page/view.php?id=12784) |  |
| 4 | [Frequency distributions](https://courseware.dceedu.org:6862/mod/page/view.php?id=13058) |  |
| 5 | Averages |  |
| 6 | Variability |  |
| 7 | Normal curves |  |
| 8 | Correlation and scatter plots |  |
| 9 | Correlation coefficient |  |
| 10 | Regression |  |

**Ex. No. : 01 WORKING WITH NUMPY ARRAYS**

**Date:**

**Aim:**

To write a program to work with Numpy arrays

**Algorithm:**

1. Start

 2. Import numpy package

 3. Create 1D, 2D and 3D arrays using numpy

 4. Perform indexing and slicing in arrays

 5. Print the desired result

 6. Stop

**Program:**

#Creating 1D array

import numpy as np  
arr = np.array([1, 2, 3, 4, 5])  
print(arr)

#Creating 2D array

import numpy as np  
arr = np.array([[1, 2, 3], [4, 5, 6]])  
print(arr)

#Creating 3D array

import numpy as np

arr = np.array([[[1, 2, 3], [4, 5, 6]], [[1, 2, 3], [4, 5, 6]]])

print(arr)

#Accessing 1D array

import numpy as np

arr = np.array([1, 2, 3, 4])

print(arr[0])

#Accessing 2D array

import numpy as np

arr = np.array([[1,2,3,4,5], [6,7,8,9,10]])

print('2nd element on 1st dim: ', arr[0, 1])

#Accessing 3D array

import numpy as np

arr = np.array([[[1, 2, 3], [4, 5, 6]], [[7, 8, 9], [10, 11, 12]]])

print(arr[0, 1, 2])

#Slicing 1D array

import numpy as np

arr = np.array([11, 12, 13, 14, 15, 16, 17])

print(arr[1:5])

#Slicing 2D array

import numpy as np

arr = np.array([[1, 2, 3, 4, 5], [6, 7, 8, 9, 10]])

print(arr[1, 1:4])

**Output:**

**#Creating 1D array**

**[1 2 3 4 5]**

**#Creating 2D array**

**[[1 2 3]**

**[4 5 6]]**

**#Creating 3D array**

**[[[1 2 3]**

**[4 5 6]]**

**[[1 2 3]**

**[4 5 6]]]**

**#Accessing 1D array**

**1**

**#Accessing 2D array**

**2**

**#Accessing 3D array**

**6**

**#Slicing 1D array**

**[12 13 14 15]**

**#Slicing 2D array**

**[7 8 9]**

**Result:**

Thus the Python Program to work with numpy arrays is executed successfully and the output is verified.

**Ex. No. : 02 WORKING WITH DATAFRAMES**

**Date:**

**Aim:**

To write a program to work with Pandas data frames

**Algorithm:**

 1. Start

 2. Import pandas package

 3. Create lists and dictionaries

 4. Create data frames using lists and dictionaries

 5. Print the data frame

 6. Stop

**Program:**

# Creating DataFrame from list

import pandas as pd

data = [['tom', 10], ['nick', 15], ['juli', 14]]

df = pd.DataFrame(data, columns = ['Name', 'Age'])

print(df)

# Creating DataFrame from dictionary

import pandas as pd

data = {'Name':['Tom', 'nick', 'krish', 'jack'],'Age':[20, 21, 19, 18]}

df = pd.DataFrame(data)

print(df)

# Creating DataFrame with explicit index

import pandas as pd

data = {'Name':['Tom', 'Jack', 'nick', 'juli'],'marks':[99, 98, 95, 90]}

df = pd.DataFrame(data, index =['rank1','rank2','rank3','rank4'])

print(df)

# Creating DataFrame from list of dicts

import pandas as pd

data = [{'a': 1, 'b': 2, 'c':3},{'a':10, 'b': 20, 'c': 30}]

df = pd.DataFrame(data)

print(df)

# Add records to dataframe using the .loc function

import pandas as pd

df = pd.DataFrame(columns = ['year','make','model'])

df.loc[0] = [2014,"toyota","corolla"]

df.loc[1] = [2018,"honda","civic"]

print(df)

**Output:**

**# Creating DataFrame from list**

**Name Age**

**0 tom 10**

**1 nick 15**

**2 juli 14**

**# Creating DataFrame from dictionary**

**Name Age**

**0 Tom 20**

**1 nick 21**

**2 krish 19**

**3 jack 18**

**# Creating data frame with explicit index**

**Name marks**

**rank1 Tom 99**

**rank2 Jack 98**

**rank3 nick 95**

**rank4 juli 90**

**# Creating DataFrame from list of dicts**

**a b c**

**0 1 2 3**

**1 10 20 30**

**# Add records to dataframe using the .loc function**

**year make model**

**0 2014 toyota corolla**

**1 2018 honda civic**

**Result:**

Thus the Python Program to work with dataframes is executed successfully and the output is verified.

**Ex. No. : 03 BASIC PLOTS USING MATPLOTLIB**

**Date:**

**Aim:**

To write a program to create basic plots using matplotlib

**Algorithm:**

 1. Start

 2. Import matplotlib package

 3. Create various plots using various functions

 4. Display the created plot

 5. Stop

**Program:**

#Line plot

import matplotlib.pyplot as plt  
 x = [10, 20, 30, 40]  
 y = [20, 30, 40, 50]  
 plt.plot(x, y)  
 plt.title("Simple Plot")  
 plt.ylabel("y-axis")  
 plt.xlabel("x-axis")  
 plt.show()

#Histogram

import matplotlib.pyplot as plt

x = [1, 2, 3, 4, 5, 6, 7, 4]

plt.hist(x, bins = [1, 2, 3, 4, 5, 6, 7])

plt.title("Histogram")

plt.legend(["bar"])

plt.show()

#Scatter plot

import matplotlib.pyplot as plt   
x = [3, 1, 3, 12, 2, 4, 4]  
y = [3, 2, 1, 4, 5, 6, 7]  
plt.scatter(x, y)  
plt.legend("A")  
plt.title("Scatter chart")  
plt.show()

#Pie chart

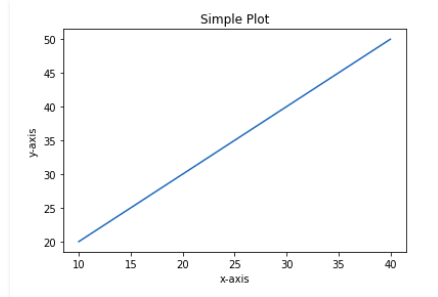
import matplotlib.pyplot as plt  
import numpy as np  
y = np.array([35, 25, 25, 15])  
mylabels = ["Apples", "Bananas", "Cherries", "Dates"]  
plt.pie(y, labels = mylabels)  
plt.show()

#Bar chart

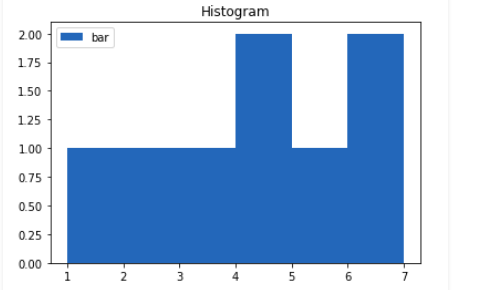
import matplotlib.pyplot as plt  
import numpy as np  
x = np.array(["A", "B", "C", "D"])  
y = np.array([3, 8, 1, 10])  
plt.bar(x,y)  
plt.show()

**Output:**

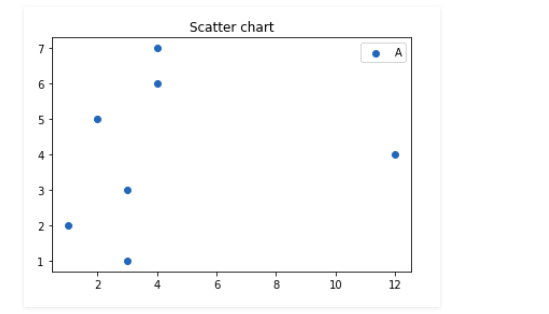
#Line plot

****

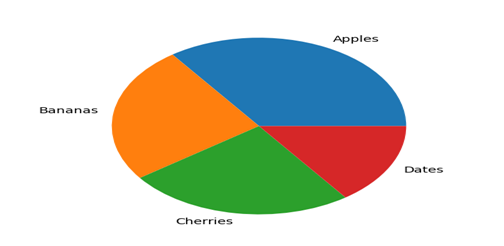
#Histogram



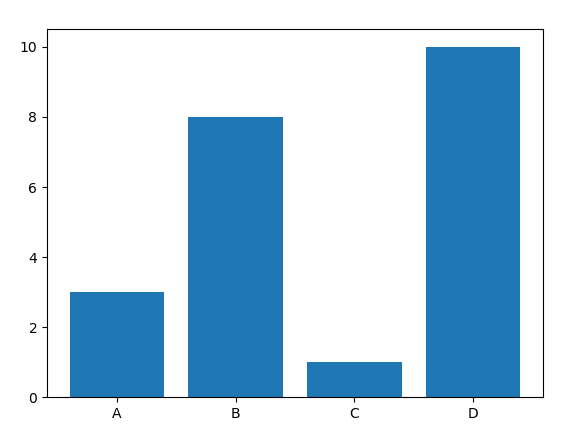
#Scatter plot



#Pie chart



#Bar chart

****

**Result:**

Thus the Python Program to create basic plots is executed successfully and the output is verified.

**Ex. No. : 04 FREQUENCY DISTRIBUTIONS**

**Date:**

**Aim:**

To write a program for frequency distributions

**Algorithm:**

 1. Start

 2. Import pandas

 3. Import pyplot from matplotlib

 4. Create a dataframe

 5. Create histograms for frequency distribution

 6. Show the plot

 7. Stop

**Program:**

#Frequency distribution for marks

import pandas as pd

import matplotlib.pyplot as plt

d={'Maths':[90,75,68,66,72,50,45],'English':[95,74,60,63,79,80,55],'Science':[60,85,58,76,52,70,65],'Names':['Avni','Bharathi','Dadlin','Irfan','Karan','Mano','Ranjit']}

df=pd.DataFrame(d)

plt.hist(df['English'])

plt.xlabel('Marks')

plt.ylabel('Count')

plt.show()

#Frequency distribution with bins

import matplotlib.pyplot as plt

x=[1,1,2,2,3,3,4,4,5,5,6,7,8,8,10,10,15,16,17,18,20,25,28,30,35,35,37,40]

plt.hist(x,bins=[0,10,20,30,40,50])

plt.show()

#Frequency distribution with colors

import pandas as pd

import matplotlib.pyplot as plt

d={'Maths':[90,75,68,66,72,50,45],'English':[95,74,60,63,79,80,55],'Science':[60,85,58,76,52,70,65],'Names':['Avni','Bharathi','Dadlin','Irfan','Karan','Mano','Ranjit']}

df=pd.DataFrame(d)

df['Maths'].plot(kind='hist',bins=[40,60,80,100],color='brown')

plt.xlabel('Marks')

plt.ylabel('Count')

plt.show()

#Frequency distribution with edge color

import pandas as pd

import matplotlib.pyplot as plt

d={'Maths':[90,75,68,66,72,50,45],'English':[95,74,60,63,79,80,55],'Science':[60,85,58,76,52,70,65],'Names':['Avni','Bharathi','Dadlin','Irfan','Karan','Mano','Ranjit']}

df=pd.DataFrame(d)

df['Maths'].plot(kind='hist',bins=4,color='red',edgecolor='black',linewidth=2)

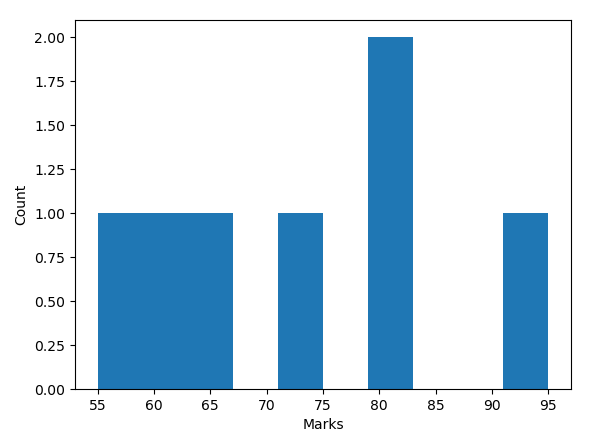
plt.xlabel('Marks')

plt.ylabel('Count')

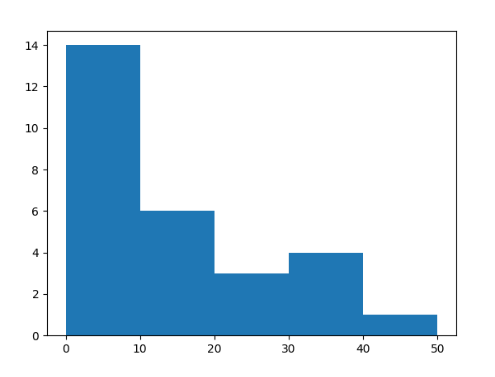
plt.show()

**Output:**

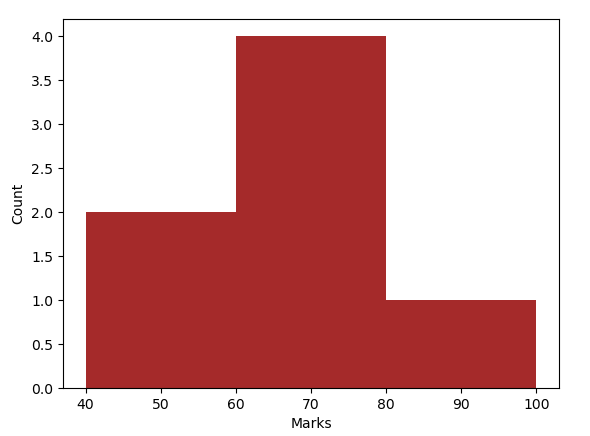
#Frequency distribution for marks



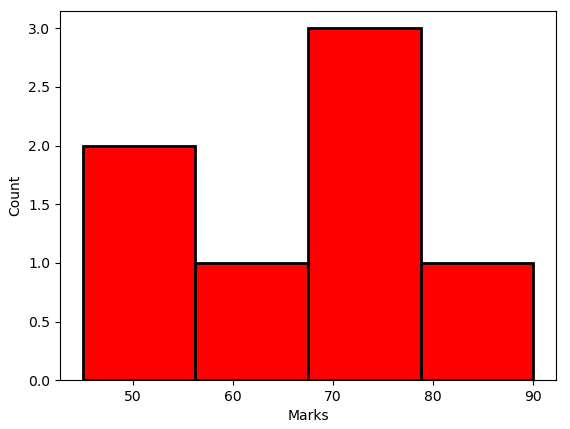
#Frequency distribution with bins



#Frequency distribution with colors



#Frequency distribution with edge color



**Result:**

Thus the Python Program for frequency distributions is executed successfully and the output is verified.

**Ex. No. : 05 AVERAGES – MEAN, MEDIAN AND MODE**

**Date:**

**Aim:**

To write a program to find mean, median and mode

**Algorithm:**

1. Start

 2. Import numpy and scipy packages

 3. Use functions mean() ,median() and mode()

 4. Print the desired result

 5. Stop

**Program:**

#Mean

import numpy as np

s=[2,4,55,6,7,7,7,6,78]

x=np.mean(s)

print(x)

#Median

import numpy as np

s=[2,4,55,6,7,7,7,6,78]

x=np.median(s)

print(x)

#Mode

import scipy.stats as st

s=[2,4,55,6,7,7,7,6,78]

x=st.mode(s)

print(x)

**Output:**

19.11111111111111

7.0

ModeResult(mode=array([7]), count=array([3]))

**Result:**

Thus the Python Program for averages is executed successfully and the output is verified.