**DATA SCIENCE LAB MANUAL**

1. **Consider the following data of three cricket players in 10 innings T20 Match**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Player** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| **Cricketer1** | **25** | **10** | **55** | **45** | **55** | **78** | **55** | **0** | **49** | **10** |
| **Cricketer2** | **47** | **62** | **78** | **45** | **100** | **20** | **100** | **0** | **80** | **10** |
| **Cricketer3** | **80** | **17** | **7** | **10** | **45** | **79** | **75** | **75** | **80** | **42** |

1. **Find Whose average is better.**
2. **What is the middlemost value of each player?**
3. **Whose most frequent value is good.**
4. **Draw a simple plot to show performance of players.**

Solution:

#Cricket Player Performance Analysis  
import statistics as st  
import matplotlib.pyplot as pt  
import tabulate  
Matches=[1,2,3,4,5,6,7,8,9,10]  
Player1=[25,10,55,45,55,78,55,0,49,10]  
Player2=[47,62,78,45,100,20,100,0,80,10]  
Player3=[80,17,7,10,45,79,75,75,80,42]  
#Player1 Summary  
print("Player1 Mean = ",st.mean(Player1))  
print("Player1 Median = ",st.median(Player1))  
print("Player1 Mode = ",st.mode(Player1))  
#Player2 Summary  
print("Player2 Mean = ",st.mean(Player2))  
print("Player2 Median = ",st.median(Player2))  
print("Player2 Mode = ",st.mode(Player2))  
#Player3 Summary  
print("Player3 Mean = ",st.mean(Player3))  
print("Player3 Median = ",st.median(Player3))  
print("Player3 Mode = ",st.mode(Player3))  
#Performance plot  
pt.plot(Matches,Player1)  
pt.plot(Matches,Player2)  
pt.plot(Matches,Player3)  
pt.title("Cricket Player Performance")  
pt.xlabel("Matches")  
pt.ylabel("Scores")  
pt.legend(["Player1","Player2","Player3"])  
pt.show()

OUTPUT:

Player1 Mean = 38.2

Player1 Median = 47.0

Player1 Mode = 55

Player2 Mean = 54.2

Player2 Median = 54.5

Player2 Mode = 100

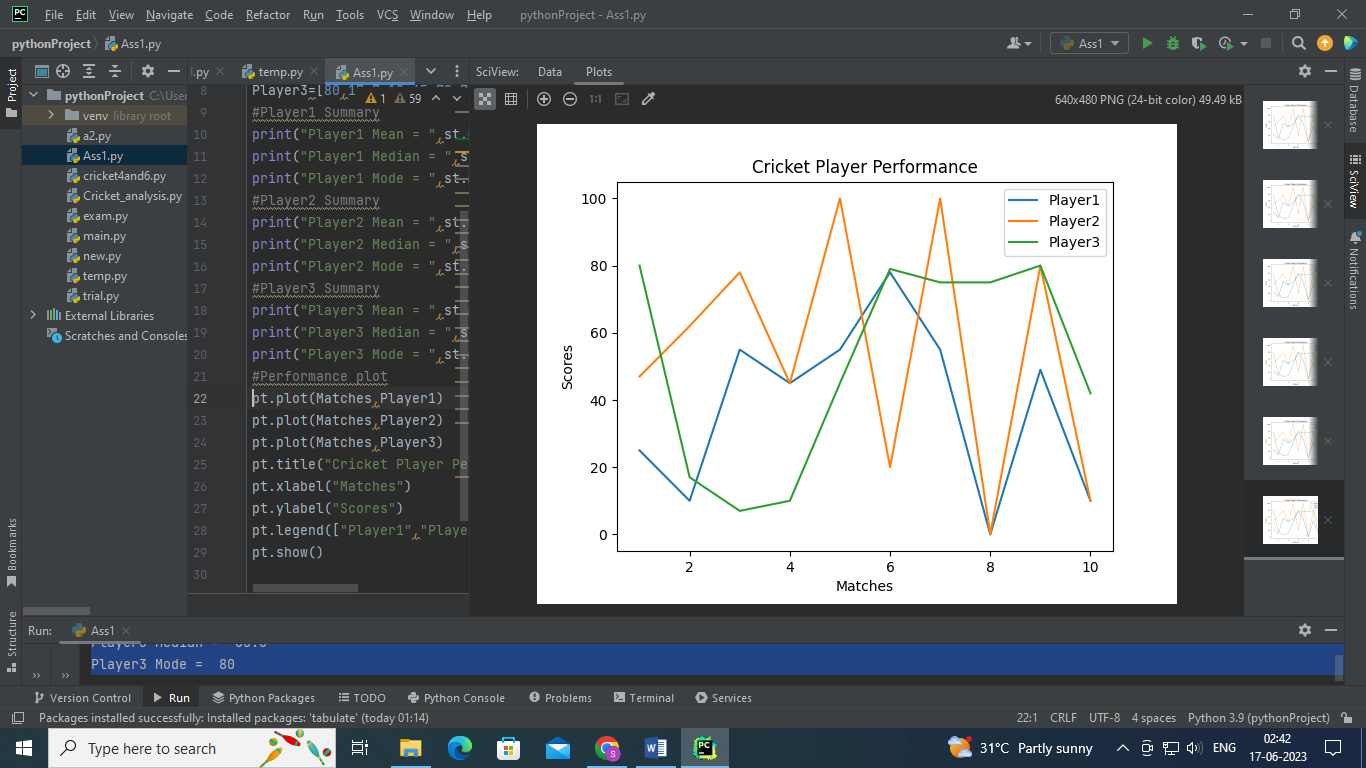
Player3 Mean = 51

Player3 Median = 60.0

Player3 Mode = 80

Analysis

1. Player 2 average is better.
2. Player1 Median = 47.0, Player2 Median = 54.5, Player3 Median = 60.0
3. Player2
4. Draw a simple plot to show performance of players.



1. Consider Insurance Dataset and analyze following
2. Count Number of Male and Female
3. What is average age of peoples.
4. Display simple bar plot Gender wise

Solution:

import pandas as pd  
import openpyxl  
import statistics as st  
import matplotlib.pyplot as pt  
data = pd.read\_csv("E:\Data Science with Python\DataSet\insurance.csv")  
print(data)  
#Analysis genderwise  
ls=data['sex'].tolist()  
y1=ls.count('female')  
y2=ls.count('male')  
print("female Count = ",y1)  
print("male Count = ",y2)  
  
#Aveage age of customers  
avgage=data['age'].tolist()  
print("Average Age= %.2f " % st.mean(avgage))  
  
#Display Histogram genderwise  
x=["FEMALE","MALE"]  
y=[y1,y2]  
pt.bar(x,y)  
pt.title("Genderwise Insurance Data")  
pt.xlabel("Gender")  
pt.ylabel("Count")  
pt.show()

Analysis:

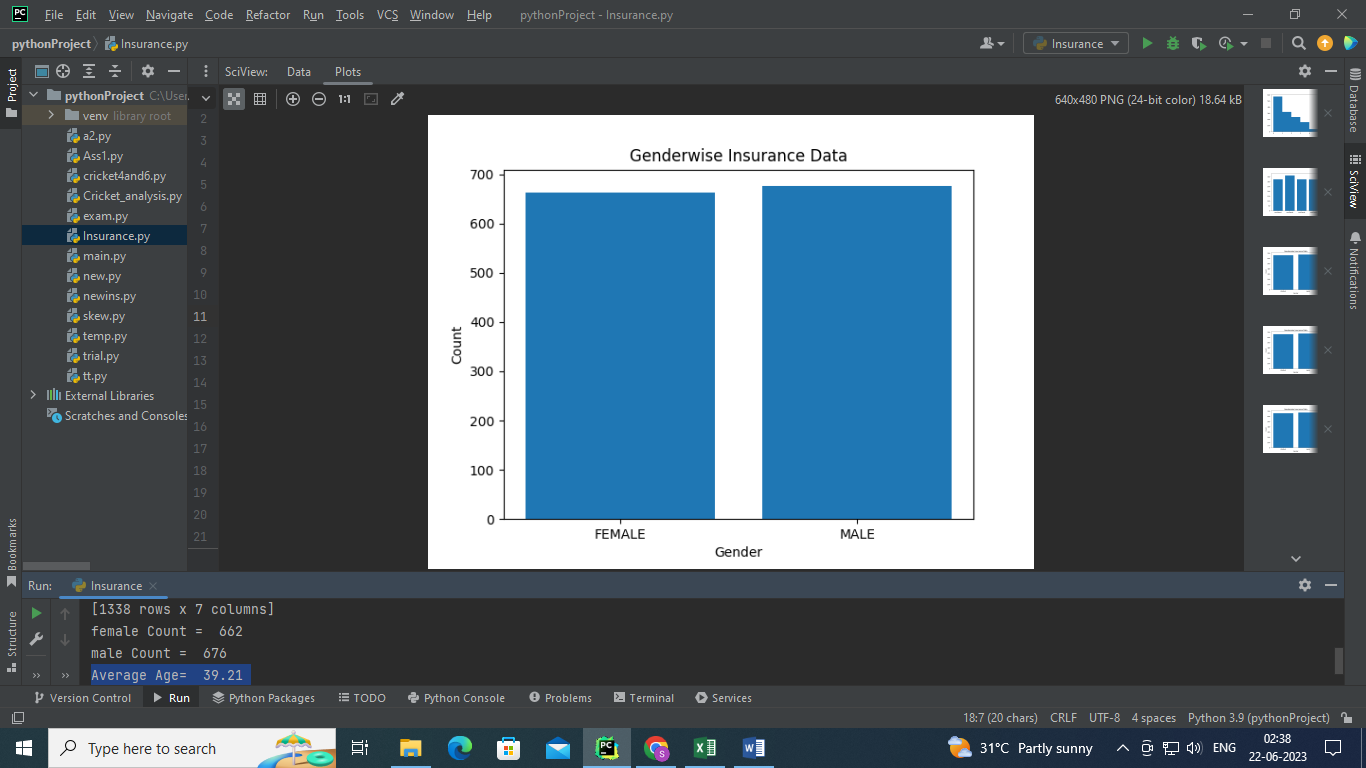
a)

female Count = 662

male Count = 676

b) Average Age= 39.21

c)



1. **Consider Insurance Dataset and analyze data region wise. Also display a simple bar chart region wise.**

**Solution:**

import pandas as pd  
import openpyxl  
import matplotlib.pyplot as pt  
data = pd.read\_csv("E:\Data Science with Python\DataSet\insurance.csv")  
print(data)  
  
#Regionwise count  
region=data['region'].tolist()  
output=[]  
for x in region:  
 if x not in output:  
 output.append(x)  
print(output)  
y1=region.count('southwest')  
y2=region.count('southeast')  
y3=region.count('northwest')  
y4=region.count('northeast')  
print("Southwest count= ",y1)  
print("southeast count= ",y2)  
print("northwest count= ",y3)  
print("northeast count= ",y4)  
pt.title("Regionwise Count")  
pt.xlabel("Region")  
pt.ylabel("Count")  
y=[y1,y2,y3,y4]  
pt.bar(output,y)  
pt.show()

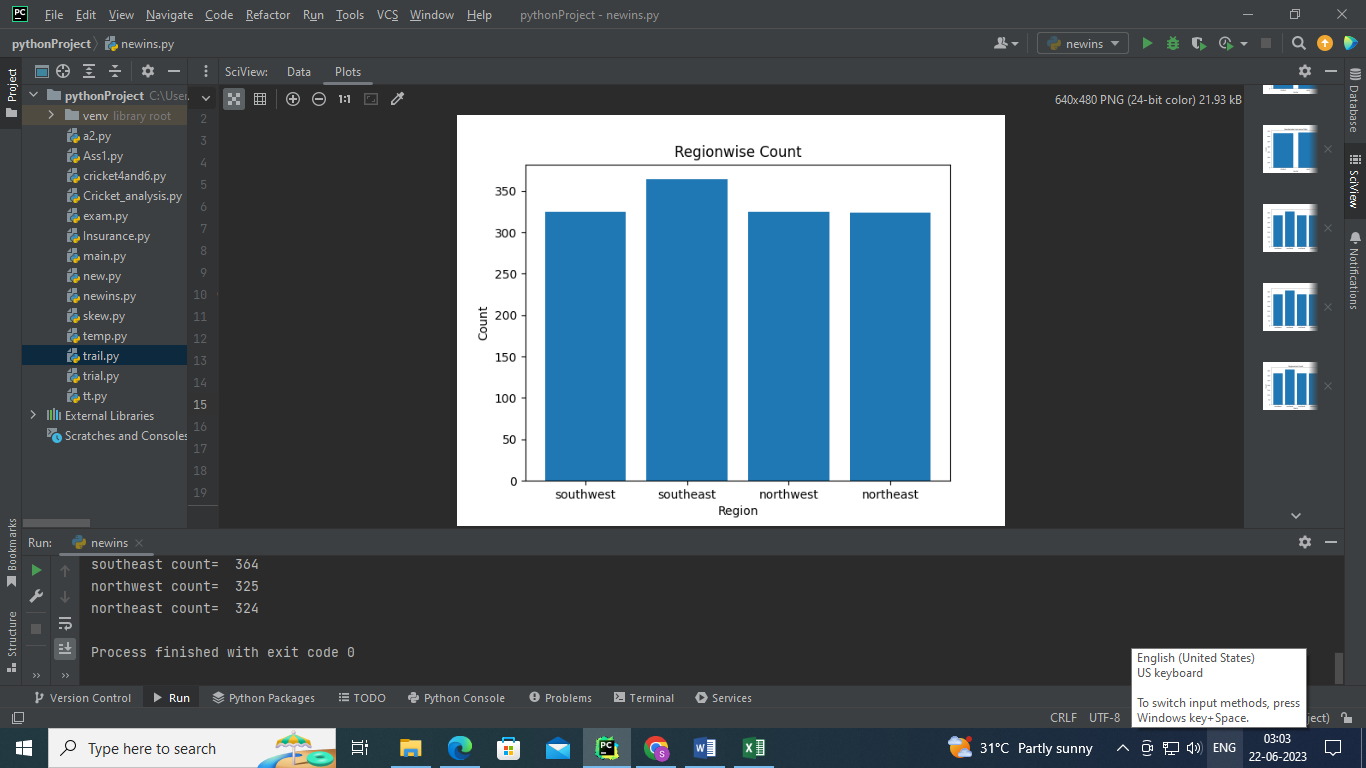
**Analysis:**

Southwest count= 325

southeast count= 364

northwest count= 325

northeast count= 324



1. Consider temperature dataset and analyze average of minimum and maximum temperature, minimum temperature, maximum temperature month wise.

Solution:

import pandas as pd  
import openpyxl  
import numpy as np  
data=pd.read\_excel("E:\\Data Science with Python\\DataSet\\belgavitemp2022.xlsx")  
print(data)  
df1 = (data.groupby(["Year", "Month"],sort=False).agg(Avg\_of\_Max\_Temp=("Max", 'mean'),  
 Max\_temp=("Max",'max'),Avg\_of\_Min\_Temp=("Min", 'mean'),Min\_temp=("Min",'min')))  
print(df1)

Analysis:

Avg\_of\_Max\_Temp Max\_temp Avg\_of\_Min\_Temp Min\_temp

Year Month

2022 January 29.290323 33 14.838710 11

February 32.535714 35 16.928571 14

March 35.451613 39 20.322581 17

April 36.666667 39 22.300000 19

May 33.838710 38 21.612903 19

June 31.533333 36 21.033333 20

July 28.225806 33 20.451613 19

August 28.419355 32 20.258065 19

September 29.533333 32 19.833333 18

October 29.741935 32 18.677419 14

November 30.433333 32 16.433333 11

December 29.870968 33 17.967742 14

**5. Consider following data and calculate Descriptive statistics using formulas.**

22,26,14,30,18,1135,41,12,32

Solution:

import numpy as np  
import pandas as pd  
data=[22,26,14,30,18,11,35,41,12,32]  
print("Mean = %.2f"% np.mean(data))  
print("Median = ",np.median(data))  
print("Max = ",np.max(data))  
print("Min = ",np.min(data))  
print("First Quartile =",np.quantile(data,0.25))  
print("Second Quartile = ",np.quantile(data,0.50))  
print("Third Quartile = ",np.quantile(data,0.75))  
print("20 th Percentilee = ",np.percentile(data,20))  
print("99 th Percentilee = ",np.percentile(data,99))  
print("Standard deviation = %.2f" % np.std(data))  
print("Variance = ",np.var(data))

OUTPUT:

Mean = 24.10

Median = 24.0

Max = 41

Min = 11

First Quartile = 15.0

Second Quartile = 24.0

Third Quartile = 31.5

20 th Percentilee = 13.6

99 th Percentilee = 40.46

Standard deviation = 9.83

Variance = 96.69

1. Find the Quartiles for the following Students Score data and visualize graphically.

50,50,47,97,49,3,53,42,26,74,82,62,37,15,70,27,36,35,48,52,63,64.

Solution:

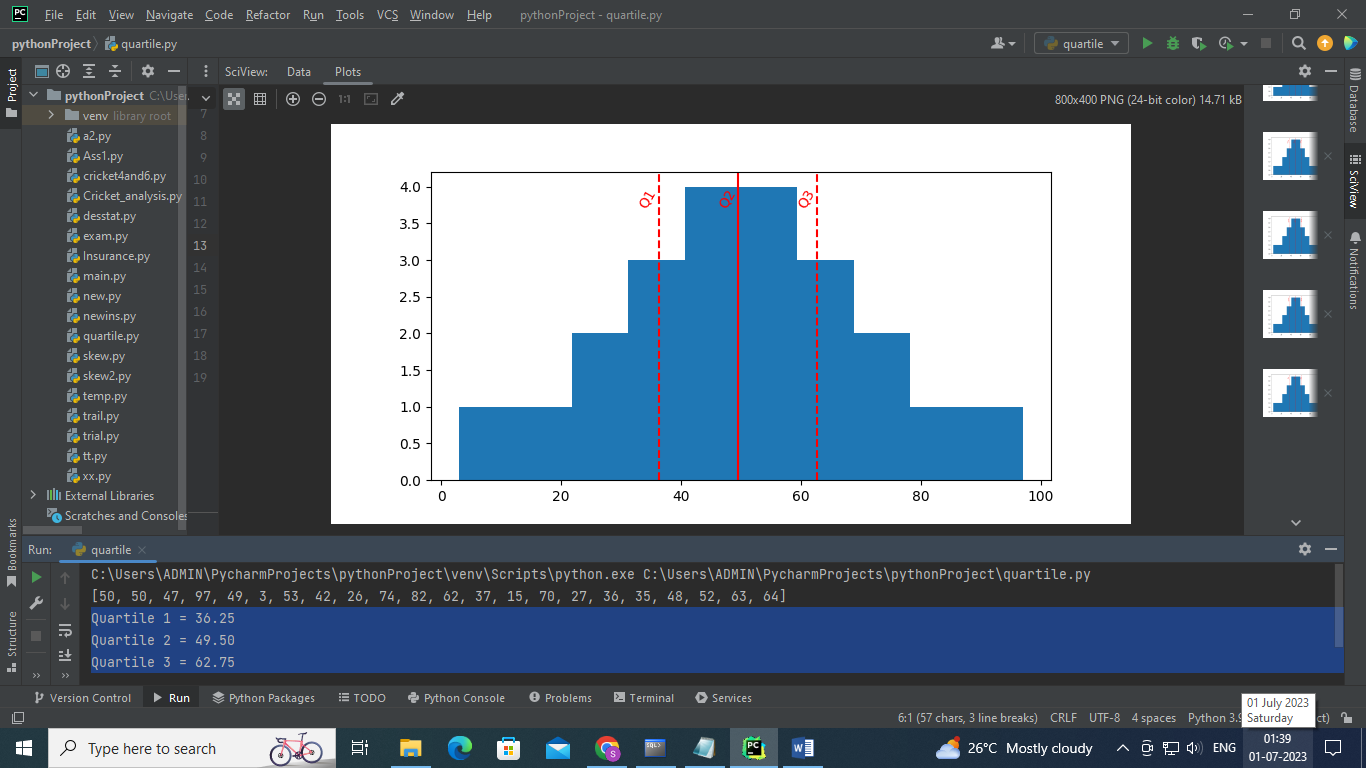
import numpy as np  
import matplotlib.pyplot as pt  
import numpy as np  
import pandas as pd  
data=[50,50,47,97,49,3,53,42,26,74,82,62,37,15,70,27,36,35,48,52,63,64]  
print(data)  
print("Quartile 1 = %.2f"%np.quantile(data,0.25))  
print("Quartile 2 = %.2f"%np.quantile(data,0.50))  
print("Quartile 3 = %.2f"%np.quantile(data,0.75))  
pt.figure(figsize=(8,4))  
pt.hist(data)  
# Vertical lines for each percentile of interest  
pt.axvline(np.quantile(data, 0.25), linestyle='--', color='red')  
pt.text(np.quantile(data, 0.25), 4, 'Q1', color='r', ha='right', va='top', rotation=60)  
pt.axvline(np.quantile(data, 0.50), linestyle='-', color='red')  
pt.text(np.quantile(data, 0.50), 4, 'Q2', color='r', ha='right', va='top', rotation=60)  
pt.axvline(np.quantile(data, 0.75), linestyle='--', color='red')  
pt.text(np.quantile(data, 0.75), 4, 'Q3', color='r', ha='right', va='top', rotation=60)  
pt.show()

OUTPUT:

Quartile 1 = 36.25

Quartile 2 = 49.50

Quartile 3 = 62.75



1. Calculate the skewness for the following data also conclude skewness

85,96,76,108,84,100,86,70,95,84

Solution

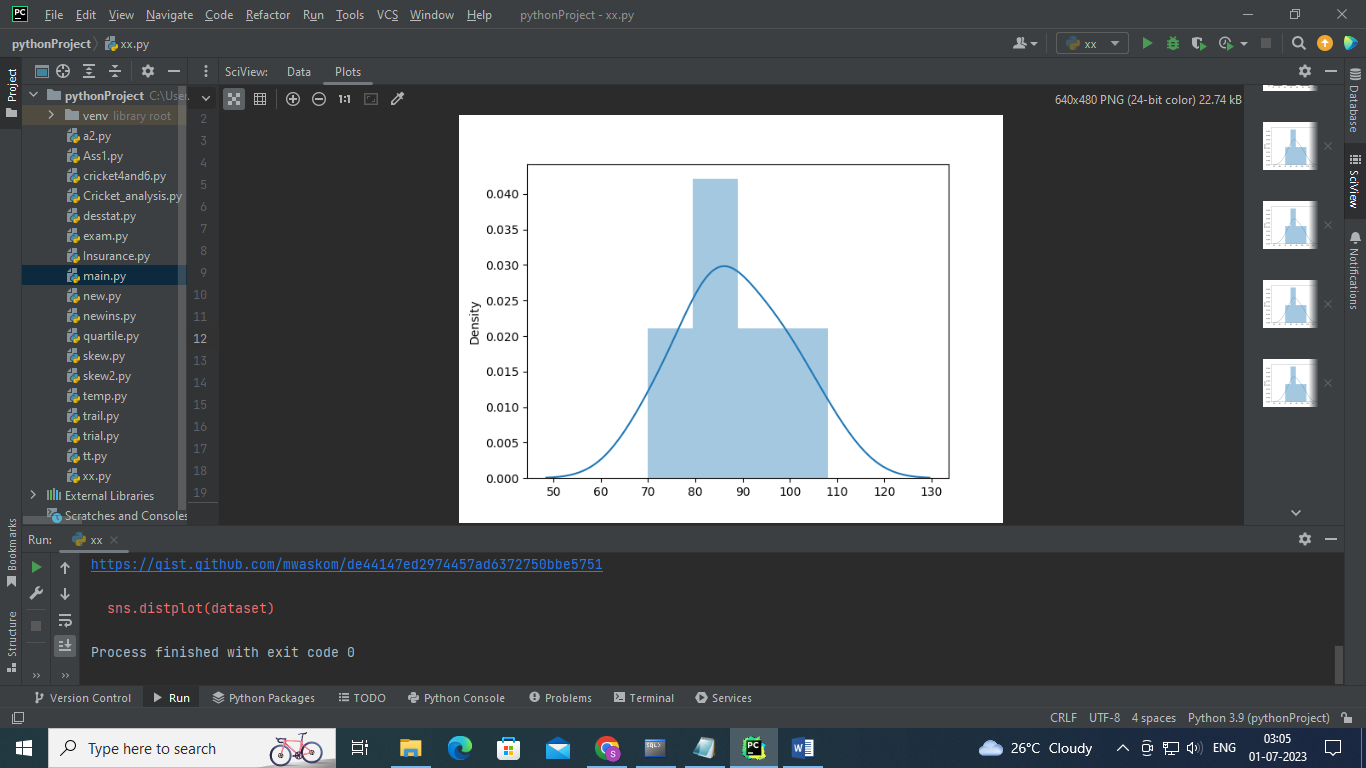
# Importing library  
import matplotlib.pyplot as pt  
import statistics as st  
import seaborn as sns  
# Creating a dataset  
dataset =[85,96,76,108,84,100,86,70,95,84]  
meandata=st.mean(dataset)  
print("Mean = %.2f"%meandata)  
modedata=st.mode(dataset)  
print("Mode = %.2f"%modedata)  
meddata=st.median(dataset)  
print("Median = %.2f"%meddata)  
# Calculate the skewness  
stddata=st.stdev(dataset)  
print("Standard Deviation =%.2f" % stddata)  
sk=(meandata-modedata)/stddata  
print("Skewness= %.2f" % sk)  
sns.distplot(dataset)  
pt.show()

OUTPUT:

Mean = 88.40

Mode = 84.00

Median = 85.50



Analysis: Distribution is Positively Skewed.

1. Consider Student Performance dataset and find skewness for all subjects.

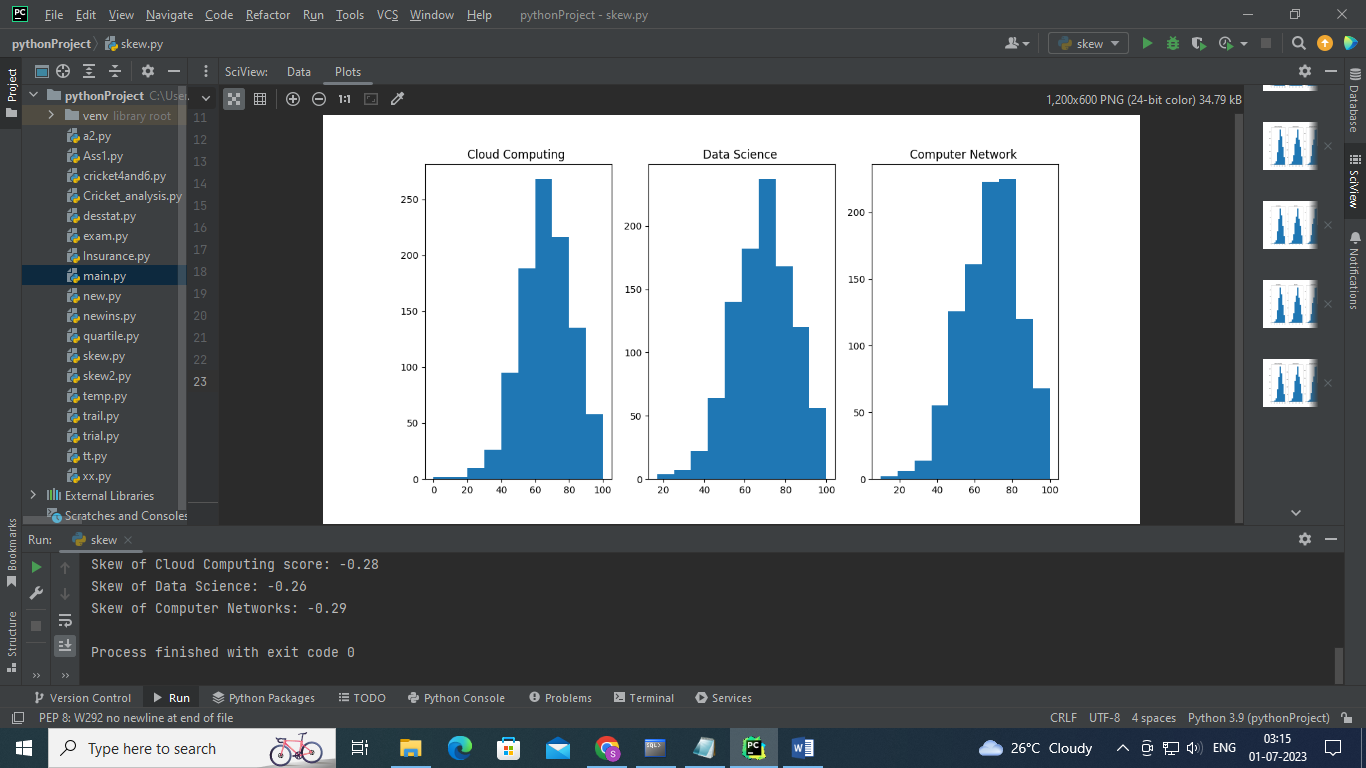
import pandas as pd  
import matplotlib.pyplot as plt  
import openpyxl  
data =pd.read\_csv("E:\Data Science with Python\DataSet\StudentsPerformance.csv")  
print(data)  
print("Skew of Cloud Computing score: %.2f"%data['Cloud Computing'].skew())  
print("Skew of Data Science: %.2f"%data['Data Science'].skew())  
print("Skew of Computer Networks: %.2f"%data['Computer Network'].skew())  
  
plt.figure(figsize = (12,6))  
plt.subplot(1, 3, 1)  
plt.hist(data['Cloud Computing'])  
plt.title('Cloud Computing ')  
  
plt.subplot(1, 3, 2)  
plt.hist(data['Data Science'])  
plt.title('Data Science ')  
  
plt.subplot(1,3,3)  
plt.hist(data['Computer Network'])  
plt.title('Computer Network ')  
  
plt.show()

OUTPUT:

Skew of Cloud Computing score: -0.28

Skew of Data Science: -0.26

Skew of Computer Networks: -0.29



Analysis:

All subjects Distribution is negatively skewed.

Maximum students score between 60-100.

1. Consider Student Performance dataset find basic statistics of data science subject using pandas describe function, calculate skewness also visualize distribution.

Solution:

import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns  
from scipy.stats import skew, skewtest, norm  
import openpyxl  
data =pd.read\_csv("E:\Data Science with Python\DataSet\StudentsPerformance.csv")  
print(data)  
print(data['Data Science'].describe())  
print("Skewness= %.2f"%data['Data Science'].skew())  
sns.distplot(data['Data Science'], fit=norm, color="r")  
plt.show()

OUTPUT:

count 1000.000000

mean 69.169000

std 14.600192

min 17.000000

25% 59.000000

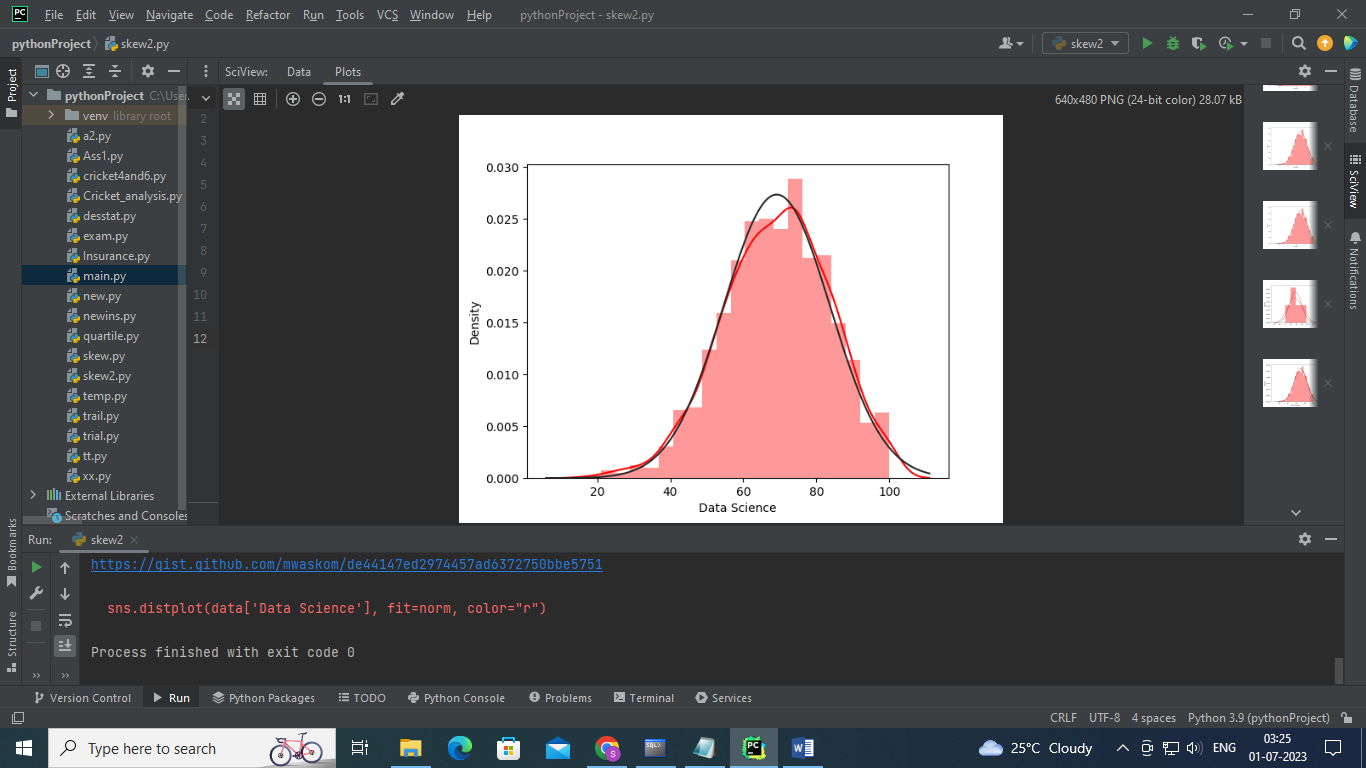
50% 70.000000

75% 79.000000

max 100.000000

Name: Data Science, dtype: float64

Skewness= -0.26



1. **Draw Regression Line for the following data. Conclude your analysis.**

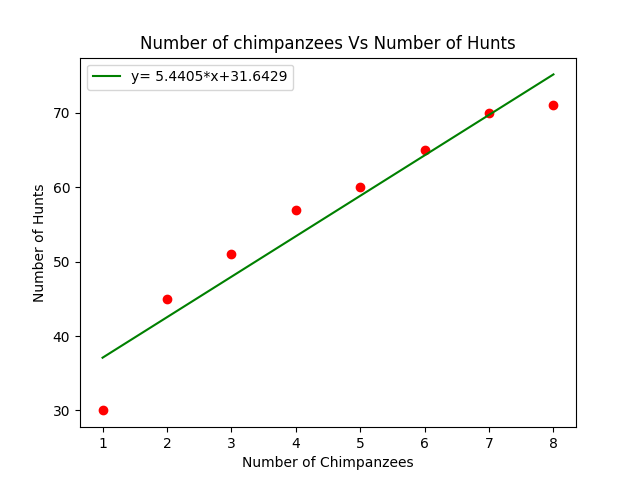
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No. of chimpanzees | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| No. of hunting | 30 | 45 | 51 | 57 | 60 | 65 | 70 | 71 |

# Import packages  
import numpy as np  
import matplotlib.pyplot as plt  
x= np.array([1,2,3,4,5,6,7,8])  
# Dependent Variable - percent of successful hunts  
y = np.array([30,45,51,57,60,65,70,71])  
n = np.size(x)  
x\_mean = np.mean(x)  
y\_mean = np.mean(y)  
b1=n \* np.sum(x\*y)-np.sum(x)\*np.sum(y)  
b2=(n \* sum(x\*x) - (np.sum(x)\*np.sum(x)))  
b=(b1/b2)  
a= y\_mean-b\*x\_mean  
print("Line Slope is : %.4f"%b)  
print("Line Intercept is: %.4f"%a)  
y\_pred=b\*x+a  
plt.scatter(x, y, color = 'red')  
plt.plot(x, y\_pred, color = 'green',label='y= 5.4405\*x+31.6429')  
plt.xlabel('Number of Chimpanzees')  
plt.ylabel('Number of Hunts')  
plt.title("Number of chimpanzees Vs Number of Hunts")  
plt.legend()  
plt.show()

OUTPUT:

Line Slope is : 5.4405

Line Intercept is: 31.6429



**Analysis:**

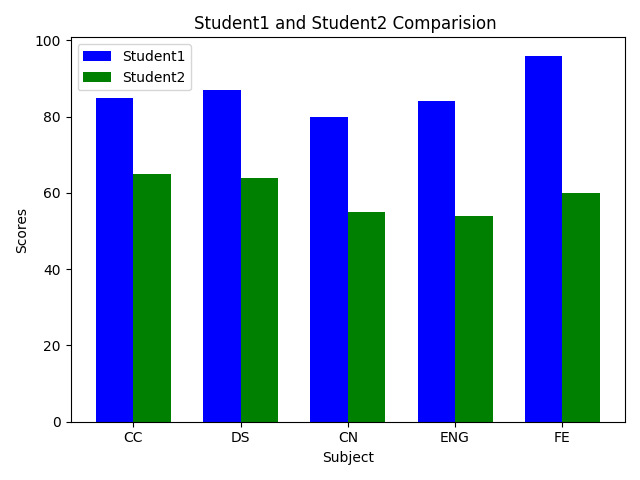
Positive Correlation exist between number of chimpanzees and number of hunts.

1. Display performance of two students in different subjects using bar graph. Also Comment on analysis.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Student | CC | DS | ENG | CN | FE |
| Student1 | 85 | 87 | 80 | 84 | 96 |
| Student2 | 65 | 64 | 55 | 54 | 60 |

import matplotlib.pyplot as plt  
import numpy as np  
Stud1=[85,87,80,84,96]  
Stud2=[65,64,55,54,60]  
# create plot  
bar\_width = 0.35  
X = np.arange(5)  
p1 = plt.bar(X, Stud1, bar\_width, color='b',label='Student1')  
# The bar of second plot starts where the first bar ends  
p2 = plt.bar(X + bar\_width, Stud2, bar\_width,color='g',label='Student2')  
plt.xlabel('Subject')  
plt.ylabel('Scores')  
plt.title('Student1 and Student2 Comparision ')  
plt.xticks(X + (bar\_width/2) , ("CC","DS","CN","ENG","FE"))  
plt.legend()  
plt.tight\_layout()  
plt.show()

OUTPUT:



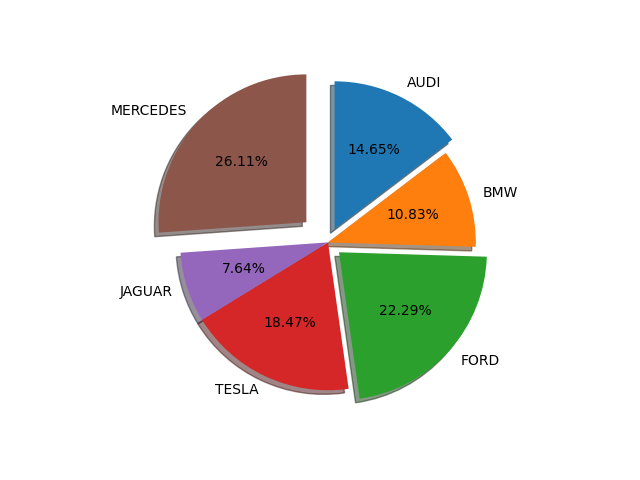
Student1 performance is good compared to student2.

1. Draw Pie chart for following data with explode, Shadow parameter.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| cars | AUDI | BMW | FORD | TESLA | JAGUAR | MERCEDES |
| data | 23 | 17 | 35 | 29 | 12 | 41 |

# Import libraries  
from matplotlib import pyplot as plt  
import numpy as np  
 # Creating dataset  
cars = ['AUDI', 'BMW', 'FORD','TESLA', 'JAGUAR', 'MERCEDES']  
data = [23, 17, 35, 29, 12, 41]  
 # Creating plot  
explode = [0.1, 0, 0.1, 0, 0,0.2]  
plt.pie(data, labels = cars,autopct='%1.2f%%',  
 explode=explode,shadow = True,startangle = 90,counterclock=False)  
 # show plot  
plt.show()

OUTPUT:

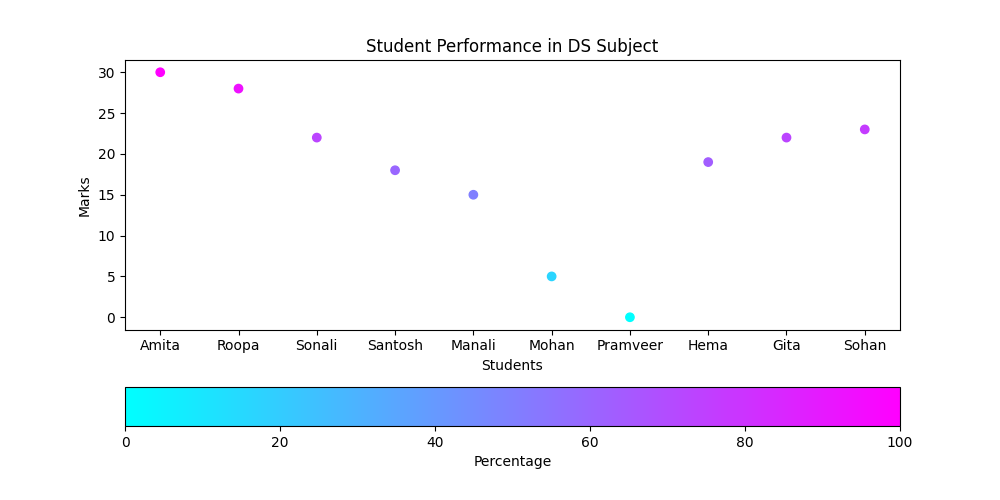


1. **Consider the following Marks data of students and draw color bar for percentage. Also analyze data. Given marks is out of 30.40% and above Passing percentage.**

marks= [30,28,22,18,15,5,0,19,22,23]

import matplotlib.pyplot as plt  
rollno= ["Amita","Roopa","Sonali","Santosh","Manali","Mohan","Pramveer","Hema","Gita","Sohan"]  
marks= [30,28,22,18,15,5,0,19,22,23]  
perls=[]  
for i in marks:  
 per="%.2f"%(i/30\*100)  
 perls.append(float(per))  
plt.figure(figsize=(10, 5))  
plt.scatter(x=rollno, y=marks, c=perls, cmap="cool")  
plt.colorbar(label="Percentage", orientation="horizontal")  
plt.title("Student Performance in DS Subject")  
plt.xlabel("Students")  
plt.ylabel("Marks")  
plt.show()

**OUTPUT:**

****

**Analysis:**

**Mohan and pramveer is failed because their percentage is between 0 to 20. Remaining 6 students Passed DS exam.**