**Q.2 A) Write a Python program for Handling Missing Value. Replace missing value of salary, age column with mean of that column.(Use Data.csv file). [5]**

import pandas as pd  
import scipy.stats  
from sklearn import preprocessing  
import matplotlib.pyplot as plt  
df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\Data.csv")  
Valuemean= df['age'].mean()  
df['age'].fillna(Valuemean, inplace= True)  
Valuemean=df['salary'].mean()  
df['salary'].fillna(Valuemean, inplace= True)  
print(df)

**Q.2 B) Write a Python program to generate a line plot of name Vs salary [5]**

**(datalineplot.py)**

import pandas as pd  
  
import matplotlib.pyplot as plt  
df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\Data.csv")  
plt.plot(df.age)  
  
*# Show the plot*plt.show()

**Q.2 C) Download the heights and weights dataset and load the dataset froma given csv file into a dataframe. Print the first, last 10 rows and random 20 rows also display shape of the dataset. [5] (plot.py)**

import pandas as pd  
df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\weight-height.csv ");  
print(df.head(10))  
print(df.tail(10))  
print(df.sample(20))

**Q.2 A)Write a Python program to create box plots to see how each feature i.e. Sepal Length, Sepal Width, Petal Length, Petal Width are distributed across the three species. (Use iris.csv dataset) [10]**

import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\Iris.csv")  
a = df[["SepalLengthCm", "SepalWidthCm", "PetalLengthCm", "PetalWidthCm"]]  
plt.boxplot(a)  
plt.show()

**Q.2 B) Write a Python program to view basic statistical details of the data (Use Heights and Weights Dataset) (stat.py)**

import statistics as st  
import pandas as pd

from pandas.api.types import is\_numeric\_dtype

df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\weight-height.csv ")  
print("Mean =", end="")  
print(st.mean(df.Height))  
print("Mode =", end="")  
print(st.mode(df.Height))  
print("Median =", end="")  
print(st.median(df.Height))  
print("Standerd Deviation = ", end="")  
print(st.pstdev(df.Height))  
print(range)

**Q.2 B) Write a Python program to print the shape, number of rows-columns, data types, feature names and the description of the data(Use User\_Data.csv) [5] (disk.py)**

import pandas as pd  
df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\User\_Data.csv")  
print(df.info())  
print(df.dtypes)  
print("number of rows",df.shape[0])  
print("number of columns",df.shape[1])

**Q.2 B) Create two lists, one representing subject names and the other representing marks obtained in those subjects. Display the data in a pie chart.**

*# Import libraries*from matplotlib import pyplot as plt  
import numpy as np  
  
*# Creating dataset*name = ['PHP', 'HTML', 'JAVA',  
 'OS', 'TCS', 'SE']  
  
marks = [23, 17, 35, 29, 12, 41]  
  
*# Creating plot*fig = plt.figure(figsize=(10, 7))  
plt.pie(marks, labels=name)  
  
*# show plot*plt.show()

**Q.2 A) Write a Python program to draw scatter plots to compare two features of the iris dataset [10]**

import matplotlib.pyplot as plt  
import pandas as pd  
df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\Iris.csv")  
plt.scatter(x='Species',  
 y='SepalLengthCm',  
 data=df)  
  
  
*# To show the plot*plt.show()

**Q.2 B) Write a Python program to create a data frame containing columns name, age , salary, department . Add 10 rows to the data frame. View the data frame. [5]**

import pandas as pd  
*#cteat and print data frame*df=pd.DataFrame(columns=['name','age','percentage'])  
df.loc[0]=['sai',20,33]  
df.loc[1]=['sai',20,33]  
df.loc[2]=['sai',20,33]  
df.loc[3]=['sai',20,33]  
df.loc[4]=['sai',20,33]  
df.loc[5]=['sai',20,33]  
df.loc[6]=['sai',20,33]  
df.loc[7]=['sai',20,33]  
df.loc[8]=['sai',20,33]  
df.loc[9]=['sai',20,33.6]  
print(df)

**Q.2 A) Generate a random array of 50 integers and display them using a line chart, scatter plot, histogram and box plot. Apply appropriate color, labels and styling options. [10]**

**Histogram:**

import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
  
*# random integers between 1 to 20*a = np.random.randint(1, 20, size=50)

plt.hist(a)  
plt.show()

**Scatterplot:**

import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
  
*# random integers between 1 to 20*x = np.random.randint(1, 20, size=50)  
y = np.random.randint(1, 20, size=50)  
plt.scatter(x, y)  
plt.show()

**Linechar:**

import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
  
*# random integers between 1 to 20*a = np.random.randint(1, 20, size=50)  
  
  
plt.plot(a, linestyle = 'dotted')  
plt.show()

**Boxplot:**

import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
  
*# random integers between 1 to 20*a = np.random.randint(1, 20, size=50)  
  
  
*# Creating plot*plt.boxplot(a)  
  
*# show plot*plt.show()

**Q.2 B) Add two outliers to the above data and display the box plot.**

*# Adding libraries*import numpy as np  
import pandas as pd  
import matplotlib.pyplot as plt  
  
*# random integers between 1 to 20*arr = np.random.randint(1, 20, size=50)  
  
*# two outliers taken*arr1 = np.append(arr, [27, 30])  
  
plt.boxplot(arr1)  
fig = plt.figure(figsize =(10, 7))  
plt.show()

**Q.2 A) Import dataset “iris.csv”. Write a Python program to create a Bar plot to get the frequency of the three species of the Iris data. [10]**

import pandas as pd  
import seaborn as sns  
import matplotlib.pyplot as plt  
df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\Iris.csv")  
sns.barplot(x='Species',  
 y='PetalLengthCm',  
 data=df)  
  
*# Show the plot*plt.show()

**Q.2 B)Write a Python program to create a histogram of the three species of the Iris data.**

**[5]**

from matplotlib import pyplot as plt  
import pandas as pd  
import numpy as np  
*# Creating dataset*df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\Iris.csv")  
plt.hist(x='Species',  
 y='SepalLengthCm',  
 data=df)  
plt.show()

**Q.2 A) Import dataset “iris.csv”. Write a Python program to create a Bar plot to get the frequency of the three species of the Iris data. [10]**

import pandas as pd  
import seaborn as sns  
import matplotlib.pyplot as plt  
df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\Iris.csv")  
sns.barplot(x='Species',  
 y='PetalLengthCm',  
 data=df)  
  
*# Show the plot*plt.show()

**Q.2 B) Write a Python program to create a histogram of the three species of the Iris data. [5]**

from matplotlib import pyplot as plt  
import pandas as pd  
import numpy as np  
*# Creating dataset*df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\Iris.csv")  
plt.hist(x='Species',  
 y='SepalLengthCm',  
 data=df)  
plt.show()

**Q.2 B) Create two lists, one representing subject names and the other representing marks obtained in those subjects. Display the data in a pie chart. [5]9**

*# Import libraries*from matplotlib import pyplot as plt  
import numpy as np  
  
*# Creating dataset*name = ['PHP', 'HTML', 'JAVA',  
 'OS', 'TCS', 'SE']  
  
marks = [23, 17, 35, 29, 12, 41]  
  
*# Creating plot*fig = plt.figure(figsize=(10, 7))  
plt.pie(marks, labels=name)  
  
*# show plot*plt.show()

**Write a Python program to perform the following tasks :**

1. **Apply OneHot coding on Country column.**

from sklearn import preprocessing  
import pandas as pd  
df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\countrydata.csv")  
enc =preprocessing.OneHotEncoder()  
onehotlable\_data =enc.fit\_transform(df[['Countey']])  
print(onehotlable\_data)

**\***

1. **Apply Label encoding on purchased column**

from sklearn.preprocessing import LabelEncoder  
import pandas as pd  
df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\countrydata.csv")  
l= LabelEncoder()  
df['purchased'] = l.fit\_transform(df['purchased'])  
print(df)

**(Data.csv have two categorical column the country column, and the purchased column).**

**[15]**

**Q.2) Write a program in python to perform following task : [15]**

**Standardizing Data (transform them into a standard Gaussian distribution with a mean of 0 and a standard deviation of 1) (Use winequality-red.csv)**

import pandas as pd  
import sklearn  
from sklearn import preprocessing as per  
from sklearn.preprocessing import StandardScaler

df= pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\winequality-red.csv", sep= ",")

*#standerdization*scaler=StandardScaler().fit(df)  
sd= scaler.transform(df)  
sd= pd.DataFrame(sd, index=df.index, columns=df.columns)  
print(sd)

**Q.2 A) Write a python program to Display column-wise mean, and median for SOCR- HeightWeight dataset. [10]**

import pandas as pd  
from pandas.api.types import is\_numeric\_dtype  
df = pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\weight-height.csv")  
for col in df.columns:  
 if is\_numeric\_dtype(df[col]):  
 print('%s:'%(col))  
  
 print('\t Mean=%2f'%df[col].mean())  
 print('\t Median=%.2f'%df[col].median())

**Q.2 B) Write a python program to compute sum of Manhattan distance between all pairs of points. [5]**

def get\_manhattan\_distance(p, q):  
 distance= 0  
 for p\_i,q\_i in zip(p, q):  
 distance += abs(p\_i-q\_i)  
 return distance  
a= (1,1)  
b=(4, 3)  
d= get\_manhattan\_distance(a, b)  
print(d)

**Q.2) Dataset Name: winequality-red.csv [15]**

**Write a program in python to perform following tasks**

**a. Rescaling: Normalised the dataset using MinMaxScaler class**

**b. Standardizing Data (transform them into a standard Gaussian distribution with a mean of 0 and**

**a standard deviation of 1)**

**c. Normalizing Data ( rescale each observation to a length of 1 (a unit norm). For this, use the**

**Normalizer class.)**

import pandas as pd  
import sklearn  
from sklearn import preprocessing as per  
from sklearn.preprocessing import StandardScaler  
from sklearn.preprocessing import Normalizer  
df= pd.read\_csv(r"C:\Users\OM\Desktop\DS slip Slutions\Dataset\winequality-red.csv", sep= ",")  
*#rescaling*scaler= per.MinMaxScaler(feature\_range=(0, 1))  
rescaleData= scaler.fit\_transform(df)  
rescaleData= pd.DataFrame(rescaleData, index=df.index, columns=df.columns)  
print(rescaleData)  
*#standerdization*scaler=StandardScaler().fit(df)  
sd= scaler.transform(df)  
sd= pd.DataFrame(sd, index=df.index, columns=df.columns)  
print(sd)  
*#Normalizing*scaler=Normalizer().fit(df)  
nd= scaler.transform(df)  
nd= pd.DataFrame(nd, index=df.index, columns=df.columns)  
print(nd)