



**CourseName:** Advanced Python Programming

**Course code:** 22CSH-623

## Experiment: 3.2

### Aim:

- a) Write a python program to implement Numpy and its related functions like ones(), zeros( ), arrange( ), reshape( ) etc
- b) Write a python program to read a csv file using Pandas and implement various in built functions. Also use the data pre-processing functions
- c) Write a python program to illustrate the use of matplotlib for visualizaing scatter plot, pie chart, and bar graph.

**Tools/Software Required:** Jupyter Notebook and Python 3.10.7

**Description:** NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

Pandas is a software library written for the Python programming language for data manipulation and analysis. In particular, it offers data structures and operations for manipulating numerical tables and time series.

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.

- A)** Write a python program to implement Numpy and its related functions like ones( ), zeros( ), arrange( ), reshape( ) etc

**CourseName:** Advanced Python Programming

**Course code:** 22CSH-623

### **Pseudo code/Algorithms/Flowchart/Steps:**

**Step 1:** import all the libraries

**Step 2:** numpy.zeros(Tuple\_values) or numpy.ones(Tuple\_values)  
or numpy.arange(Number).reshape(dimensions)

**Step 3:** Print(variable)

### **Implementation:**

```
import numpy
print(numpy.zeros((5,9)), "\n")
print(numpy.ones((4,6)), "\n")
arr = numpy.arange(12)
print(arr.reshape(4, 3))
```

### **Output:**

```
[[0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0.]
 [0. 0. 0. 0. 0. 0. 0. 0. 0.]]
```

```
[[1. 1. 1. 1. 1. 1.]
 [1. 1. 1. 1. 1. 1.]
 [1. 1. 1. 1. 1. 1.]
 [1. 1. 1. 1. 1. 1.]]
```

```
[[ 0  1  2]
 [ 3  4  5]
 [ 6  7  8]
 [ 9 10 11]]
```



**CourseName:** Advanced Python Programming

**Course code:** 22CSH-623

**b) Write a python program to read a csv file using Pandas and implement various in built functions. Also use the data pre-processing functions**

**Pseudo code/Algorithms/Flowchart/Steps:**

**STEP1:** Import the Pandas library.

**STEP 2:** Use the read\_csv function from Pandas to read the CSV file into a Pandas dataframe.

**STEP 3:** If necessary, use various in-built functions and methods from Pandas to manipulate and analyze the data in the dataframe.

**STEP 4:** If needed, use data preprocessing functions from libraries such as scikit-learn to prepare the data for further analysis or modelling

**Implementation:**

```
import pandas as pd

df =
pd.read_csv(r"https://raw.githubusercontent.com/datasciencedojo/datasets/master/titanic.csv")

print(df)

print(df.describe())

from sklearn.preprocessing import StandardScaler

scaler = StandardScaler()

df_scaled = scaler.fit_transform(df.iloc[:,0:2])
```



CourseName: Advanced Python Programming

Course code: 22CSH-623

## Output:

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked	
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

	PassengerId	Survived	Pclass	Age	SibSp	Parch	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.308642	29.699118	0.523008	0.381594	32.204208
std	257.353842	0.486592	0.836071	14.526497	1.102743	0.806057	49.693429
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	2.000000	20.125000	0.000000	0.000000	7.910400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	668.500000	1.000000	3.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	8.000000	6.000000	512.329200

```
array([[ -1.73010796, -0.78927234],  
       [ -1.72622007,  1.2669898 ],  
       [ -1.72233219,  1.2669898 ],  
       ...,  
       [  1.72233219, -0.78927234],  
       [  1.72622007,  1.2669898 ],  
       [  1.73010796, -0.78927234]])
```



**CourseName:** Advanced Python Programming

**Course code:** 22CSH-623

**c) Write a python program to illustrate the use of matplotlib for visualizaing scatter plot, pie chart, and bar graph.**

### **Pseudo code/Algorithms/Flowchart/Steps:**

**STEP 1:** Import the matplotlib.pyplot module and assign it to a variable (e.g., plt).

**STEP 2:** Prepare the data for the chart. This usually involves creating a list or array of x-values and a corresponding list or array of y-values.

**STEP 3:** Use the plot function of the pyplot module to plot the data. For example, plt.plot(x\_values, y\_values).

**STEP 4:** Customize the appearance of the chart by setting various attributes of the pyplot module, such as the title, axis labels, and tick marks.

**STEP 5 :** Use the show function of the pyplot module to display the chart.

### **Implementation:**

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
df = pd.read_csv("Datasets/NIFTY50.csv")
plt.plot(df["Close"], 'go--', linewidth=2, markersize=5, label = "Stock price")
plt.xlabel("Days")
plt.ylabel("Stock Price")
plt.title("Nifty 50 Chart")
plt.legend(loc = 1)
plt.show()
df = pd.read_csv("Datasets/Cancer.csv")
df = df["diagnosis"].value_counts()
keys = list(df.keys())
```

**CourseName:** Advanced Python Programming

**Course code:** 22CSH-623

```
values = list(df.values)
plt.bar(keys, values, color = ["g","r"], width = 0.6, alpha = 0.9)
plt.show()
df = df["diagnosis"].value_counts()
df.plot(kind = "pie")
```

**Output:**

