



School: SDET Campus: \_\_\_\_\_

Academic Year 2024-25 Subject Name: DAVP Subject Code: \_\_\_\_\_

Semester: 1 Program: E-TECH Branch: CSE Specialization: CN

Date: \_\_\_\_\_

## Applied and Action Learning (Learning by Doing and Discovery)

Name of the Experiment:

### \* Coding Phase: Pseudo Code / Flow Chart / Algorithm

#### Pseudo Code for scatter plot

- 1) import matplotlib.pyplot as plt
- 2) Create a list of x & y
- 3) Set title for plot
- 4) Label the x-axis & y-axis
- 5) Plot the scatter plot with color & marker
- 6) Display the plot

#### Pseudo Code for line plot

- 1) import matplotlib.pyplot as plt
- 2) Create a list of x and y
- 3) Plot the data with specified marker, color, & line style
- 4) Label the x-axis & y-axis
- 5) Set the title of plot
- 6) Display the plot

#### Pseudo Code for histogram

- 1) import matplotlib.pyplot as plt
- 2) Create a list data
- 3) Create a histogram with color and edge colour
- 4) Set the title of the plot
- 5) Label the x-axis and y-axis
- 6) Display the plot

#### Pseudo Code for Bar chart

- 1) import matplotlib.pyplot as plt
- 2) Create title for categories and value
- 3) Plot the bar chart with value and categories
- 4) Set the title of the plot
- 5) Label the x-axis & y-axis
- 6) Display the plot.

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

## \* Testing Phase: Compilation of Code (error detection)

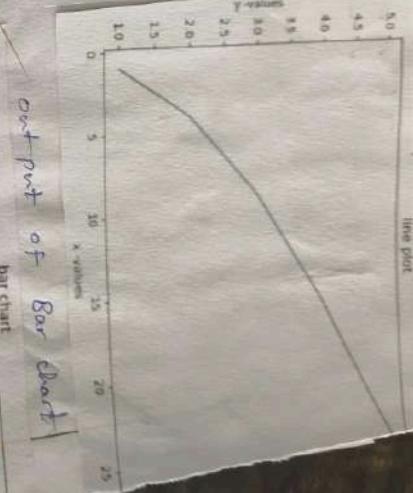
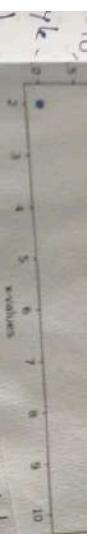
```
# line plot
import matplotlib.pyplot as plt
x = [1, 2, 3, 4, 5]
y = [1, 4, 9, 16, 25]
plt.plot(x, y, marker='o', color='black', linestyle='solid')
plt.title("Scatter plot")
plt.xlabel("x-values")
plt.ylabel("y-values")
plt.scatter(x, y, color='black', marker='o')
plt.show()

# Histogram
data = [1, 1, 2, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5]
plt.hist(data, color='gray', edgecolor='black')
plt.title('Histogram')

# Bar chart
categories=['Amn', 'Cic', 'Ca', 'Csb']
Value = [1, 2, 3, 4, 5]
plt.bar(categories, Value)
plt.title('Bar chart')
plt.xlabel('Concept')
plt.ylabel('Rubric')
plt.title('Bar chart')
plt.yticks([0, 1.5, 2.0])
plt.y_value(Values)
plt.y_value(Domains)
Total
```

Signature of t

Applied and Action Learning  
line & scatter plot



EXPERIMENT		
Rubrics	Full Mark	Marks Obtained
Concept	10	09
Planning and Execution/ Practical Simulation/ Programming	10	10
Result and Interpretation	10	10
Record of Applied and Action Learning	10	09
Viva	50	45
Total		

Signature of the Student: S. Yuganethar

Page No. ....

Name: S. Yuganethar

Regn. No.: 21118013550010

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

Signature of the Faculty:



School: SOET Campus: .....  
Academic Year: 2024 - 25 Subject Name: DevP Subject Code: .....  
Semester: I Program: E.T.E.C.L Branch: CSE Specialization: C.A.I

Date: \_\_\_\_\_

## Applied and Action Learning (Learning by Doing and Discovery)

Name of the Experiment:  
\* Coding Phase: Pseudo Code / Flow Chart / Algorithm

Pseudo code for histogram:

- 1) start
- 2) import mat plot lib
- 3) Define data list
- 4) Create histogram (with bins, color, edge color)
- 5) set title as "Histogram"
- 6) set x - label as data value as y - label as "frequency"
- 7) Show plot
- 8) end

Pseudo code for bar plot:

- ~~- 1) Start
  - 2) import mat plot lib
  - 3) Define categories and values
  - 4) Create a bar plot (color of edge color)
  - 5) set title as "Bar plot"
  - 6) set x - label as domain and y - label as values
  - 7) Show plot
  - 8) end~~

## \* Testing Phase: Compilation of Code (error detection)

```

import matplotlib.pyplot as plt
data = [1, 1, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5, 5]
# create a histogram
plt.hist(data, bins=5, color="sky blue", edgecolor="black")
plt.title("Histogram")
plt.xlabel("Data Value")
plt.ylabel("frequency")
plt.show()

```

```

import matplotlib.pyplot as plt
# create a data
categories = ("Aiml", "Ds", "ST", "Cni", "BCA")
Values = (0.5, 1, 1, 0.5, 0.5)
# create bar plot
plt.bar(categories, Values, color="sky blue", edgecolor="black")
plt.title("bar plot")
plt.xlabel("Domain")
plt.ylabel("Value")
plt.show()

```

Two sheets per experiment (10-20) to be used.  
As applicable according to the experiment.

Page No.....

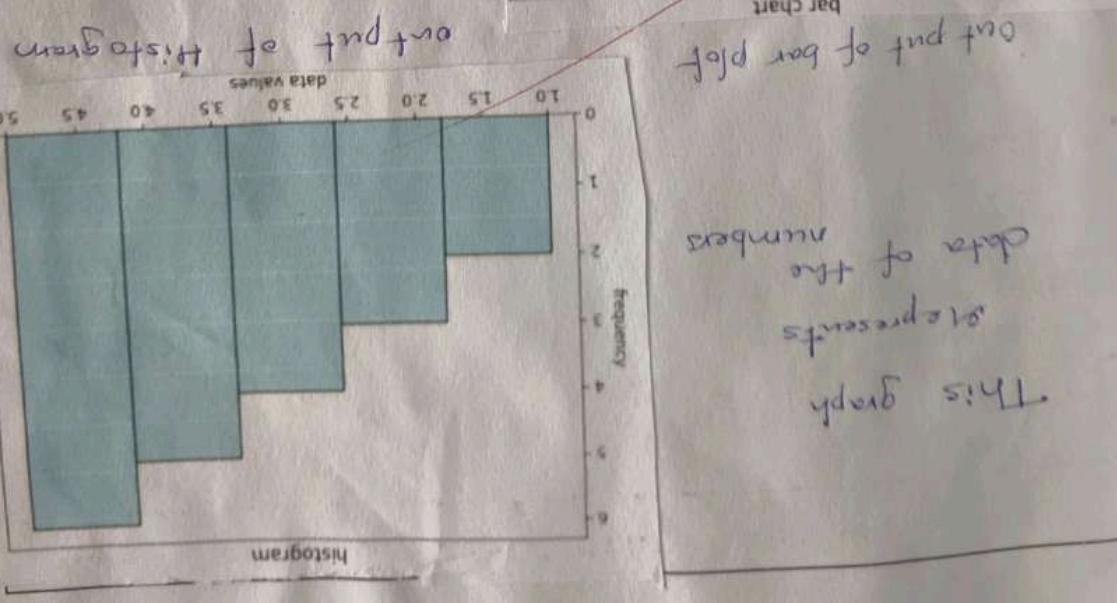
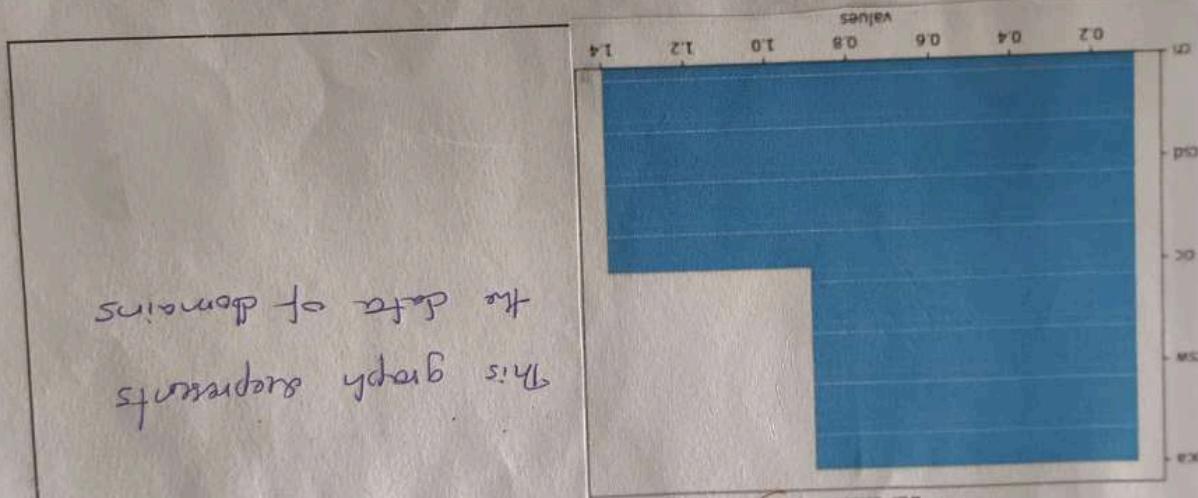
Name: S. Yugandhar  
Regn. No.: 241861350010

Signature of the Faculty:

Signature of the Student: S. Yugandhar

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	10	
Planning and Execution/ Practical Simulation/ Programming	10	69	
Result and Interpretation	10	69	
Record of Applied and Action Learning	10	99	
Viva	10	10	
Total	50	49	

## ASSESSMENT



Applied and Action Learning

action learning

black

values

0.2

0.4

0.6

0.8

1.0

1.2

1.4

0.2

0.4

0.6

0.8

1.0

1.2

1.4



School: ..... SNET ..... Campus: .....  
Academic Year: 2024 - 25 Subject Name: D.A.V.P. Subject Code: .....  
Semester: 1 Program: B.TECH! Branch: CSE Specialization: CSE  
Date: .....

## Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

### \* Coding Phase: Pseudo Code / Flow Chart / Algorithm

pseudo code for pie plot:-

- 1) start
- 2) import matplotlib lib
- 3) Define categories and values
- 4) Define explode and colour
- 5) Create pie - chart with (autopct, shadow, explode, colors)
- 6) set title as pie plot of domains
- 7) show plot
- 8) End.

pseudo code for Bar plot:-

- 1) start
- 2) Import matplotlib lib
- 3) Define Categories list
- 4) Define Value list
- 5) Create bar plot with (colour, edge color)
- 6) Set title as "Bar plot of Domain"
- 7) Label the axes
- 8) show plot
- 9) end.

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

## \* Testing Phase: Compilation of Code (error detection)

```

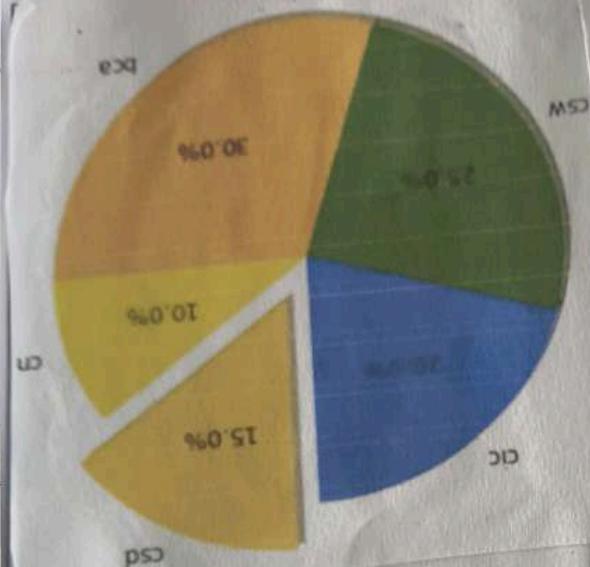
import matplotlib.pyplot as plt
categories = ["Aims", "Ps", "St", "Cn", "Bca"]
Values = [10, 20, 30, 40, 50]
Explode = [0, 0.2, 0, 0, 0]
Colors = ["yellow", "Coral", "yellowgreen", "blue", "lightgreen"]
plot pie [values, labels = categories, autopct = '%.1f %',
           shadow = "true", explode = Explode,
           colors = Colors]
plt.title ("Pie plot of domains")
plt.show ()

```

```

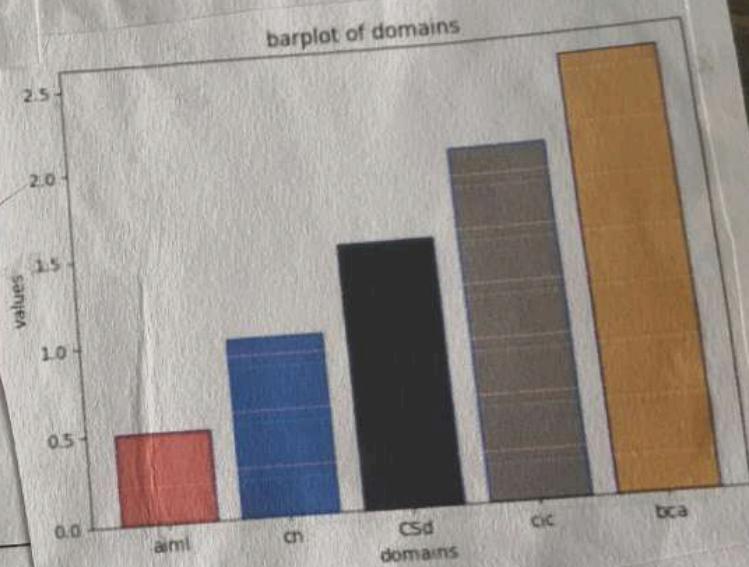
import matplotlib.pyplot as plt
categories = ["Aims", "Ps", "St", "Cn", "Bca"]
Values = [0.5, 0.5, 0.5, 2, 0.5]
plt.bar (categories, Values, color = color, edgecolor = 'blue')
plt.title ("Bar plot of Domains")
plt.xlabel ("Domain")
plt.ylabel ("Values")
plt.show ()

```



This graph represents the percentage of students present in each domain

This graph represents the no. of students present in each domain



## ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	10	
Planning and Execution/ Practical Simulation/ Programming	10	09	
Result and Interpretation	10	09	
Record of Applied and Action Learning	10	10	
Viva	10	09	
<b>Total</b>	<b>50</b>	<b>47</b>	

Signature of the Student: S.Yugandher

Name: S.Yugandher

Regn. No.: 24180135004

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

Signature of the Faculty:



School: SOET Campus:  
Academic Year: 2024 - 25 Subject Name: DAVP Subject Code:  
Semester: 1 Program: R-TECH Branch: CSE Specialization: CN

Date:

## Applied and Action Learning (Learning by Doing and Discovery)

Name of the Experiment:

### \* Coding Phase: Pseudo Code / Flow Chart / Algorithm

pseudo code for bar plot:

- 1) import matplot lib
- 2) Define categories and values
- 3) Define colour
- 4) Create a bar chart with (colour, edge colour)
- 5) Add title
- 6) Add labels
- 7) Display the chart.

pseudo code for scatter plot:

- 1) import mat plot . lib
- 2) Define x - Value
- 3) Define y - value
- 4) Create a scatter plot with (x & y values, color, marker)
- 5) Add title
- 6) Add labels
- 7) Display chart.

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

### \* Testing Phase: Compilation of Code (error detection)

```

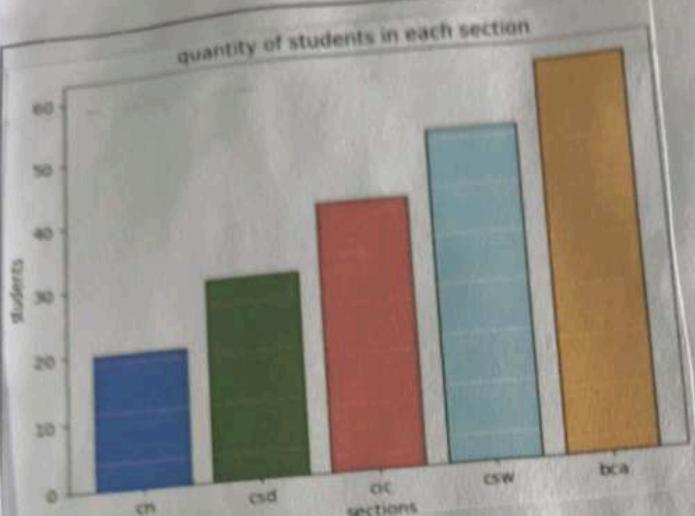
import matplotlib.pyplot as plt.
Categories = ['A', 'B', 'C', 'D', 'E']
Value = [10, 20, 30, 40, 50]
plt.bar(Categories, Value, color='blue', edgecolor='black')
plt.xlabel('Sections')
plt.ylabel('Students')
plt.title('Quantity of student in each section')
plt.show()

```

```

import matplotlib.pyplot as plt
x = ['A', 'B', 'C', 'D', 'E']
y = [2, 3, 5, 7, 11]
plt.scatter(x, y, color='red', marker='*')
plt.title('Quantity of rooms in each floor')
plt.xlabel('floors')
plt.ylabel('Rooms')
plt.show()

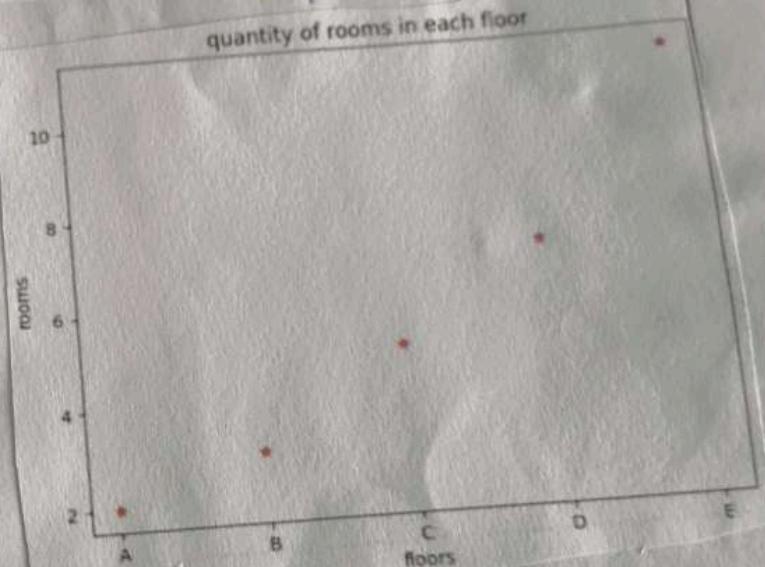
```



output of Bar plot

This graph represents  
the no. of students  
present in each section

output of scatter plot



## ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	09	
Planning and Execution/ Practical Simulation/ Programming	10	10	
Result and Interpretation	10	10	
Record of Applied and Action Learning	10	09	
Viva	50	48	
<b>Total</b>			

Signature of the Faculty:

Signature of the Student: S. Yugandher

Name: S. Yugandher

Regn. No.: 241801350040

Page No....

\*As applicable according to the experiment

Two sheets per experiment (10-20) to be



School: SOET Campus: .....  
Academic Year: 2024-25 Subject Name: D.A.V.P. Subject Code: .....  
Semester: I Program: B.TECH Branch: CSE Specialization: C.N.  
Date: .....

## Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

\* Coding Phase: Pseudo Code / Flow Chart / Algorithm

pseudocode for bubble chart :

- 1) import matplotlib.pyplot
- 2) create lists x,y, and size for co-ordinates & bubbles
- 3) plot the bubble chart using the x,y and size lists  
and specify the colour and transparency
- 4) Set title of the plot
- 5) Label the x-axis and y-axis
- 6) Display the plot.

pseudocode for Heatmap:

1. import the Scaborn, matplotlib.pyplot & numpylib
2. Create  $10 \times 10$  matrix of random data using numpy
3. plot the heatmap with the 'Coolwarm' and colour
- 4 set the title of the plot
5. Label x-axis and y-axis
6. Show the plot.

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

### \* Testing Phase: Compilation of Code (error detection)

```
import matplotlib.pyplot as plt.
```

```
x = [10, 20, 30, 40, 50]
```

```
y = [40, 30, 20, 10, 50]
```

```
size = [100, 200, 300, 400, 500]
```

```
plt.scatter(x,y, s=size, color="blue", alpha=0.5)
```

```
plt.title("Bubble chart")
```

```
plt.xlabel("x-axis")
```

```
plt.ylabel("y-axis")
```

```
plt.show()
```

```
import seaborn as sns
```

```
import matplotlib.pyplot as plt
```

```
import numpy as np
```

```
data = np.random.rand(5,5)
```

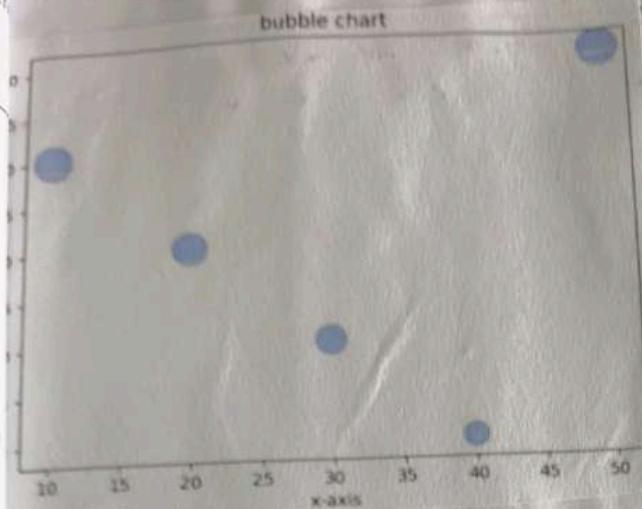
```
sns. heatmap(data, cmap='coolwarm', annot=True)
```

```
plt.title("heat map")
```

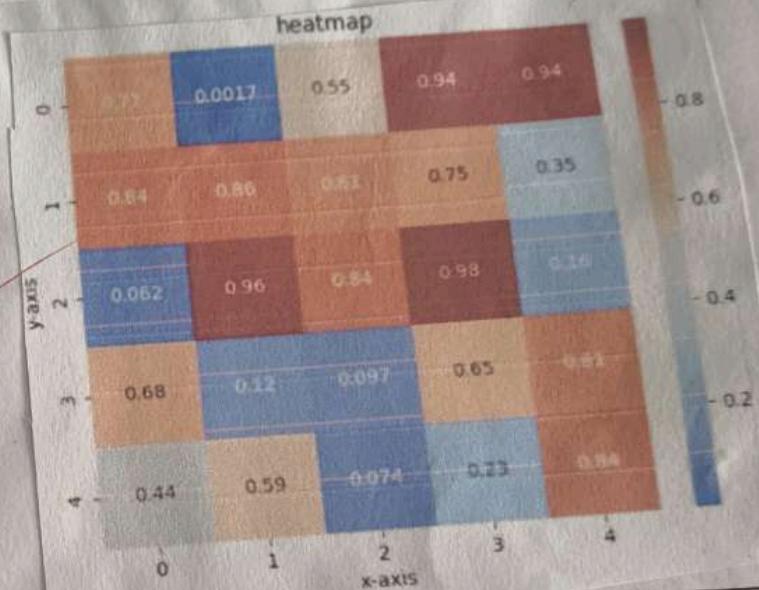
```
plt.xlabel("x-axis")
```

```
plt.ylabel("y-axis")
```

```
plt.show()
```



This graph represents  
the data in bubble chart



This graph  
represents the data  
in heatmap

## ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	10	
Planning and Execution/ Practical Simulation/ Programming	10	10	
Result and Interpretation	10	10	
Record of Applied and Action Learning	10	09	
Viva	10	10	
<b>Total</b>	<b>50</b>	<b>49</b>	

Signature of the Student: S. Yugandhar

Name: S. Yugandhar

Regn. No.: 201801350010

Page No. ....

\*As applicable according to the experiment  
Two sheets per experiment (10-20) to be submitted

Signature of the Faculty:



School: SPET Campus: Vizianagram  
Academic Year: 2024-25 Subject Name: D.A.V.P. Subject Code:  
Semester: I Program: E.TECH Branch: CSE Specialization: CN  
Date:

## Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

### \* Coding Phase: Pseudo Code / Flow Chart / Algorithm

- 1) import matplotlib library
- 2) Create lists of x, y<sub>1</sub>, y<sub>2</sub> for the data
- 3) Create first area plot b/w x, y<sub>1</sub> with color and transparency
- 4) Create Second area plot b/w x, y<sub>2</sub> by color and transparency
- 5) Set title for the plot
- 6) Label x-axis and y-axis
- 7) Add a legend to differentiable b/w two areas
- 8) Display the plot

- 1) Import matplotlib library
- 2) Create a list for labels, sizes, color and
- 3) Create pie-chart with (sizes, explode, labels, color, percentage, shadow)
- 4) Set the title for the plot
- 5) Display the plot

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

### \* Testing Phase: Compilation of Code (error detection)

```
# create area plot
import matplotlib.lib.pyplot as plt

x = [1, 3, 5, 7, 9]
y1 = [2, 4, 13, 11, 10]
y2 = [1, 3, 5, 7, 9]

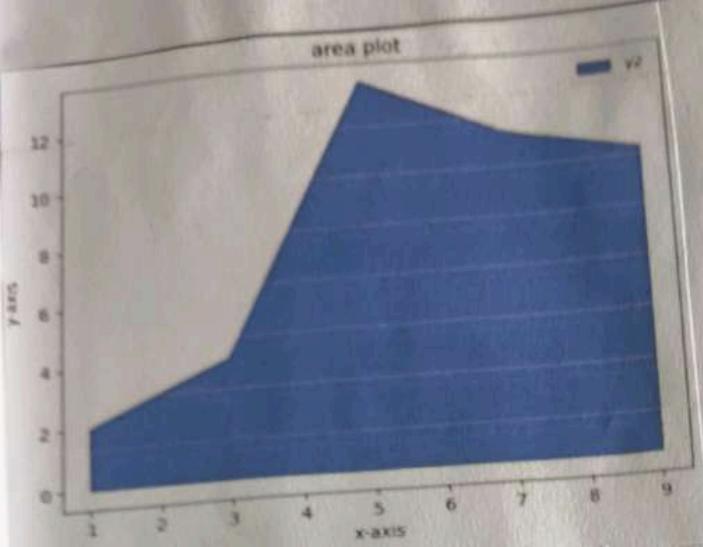
plt.fill_between(x, y1, color="indigo", alpha=1,
                 label="y1", edgecolor="black")

plt.fill_between(x, y2, color="yellow", alpha=1,
                 label="y2", edgecolor="black")

plt.title('Area plot')
plt.xlabel('x-axis')
plt.ylabel('y-axis')
plt.legend()

label = ['chocolate', 'Vanilla', 'strawberry', 'pistachio']
sizes = [30, 25, 20, 15]
colors = ['Brown', 'White', 'Pink', 'Green']
explode = [0.1, 0, 0, 0]

plt.pie(sizes, explode=explode, labels=label, colors=colors)
plt.title('flavors of ice cream')
plt.show()
```

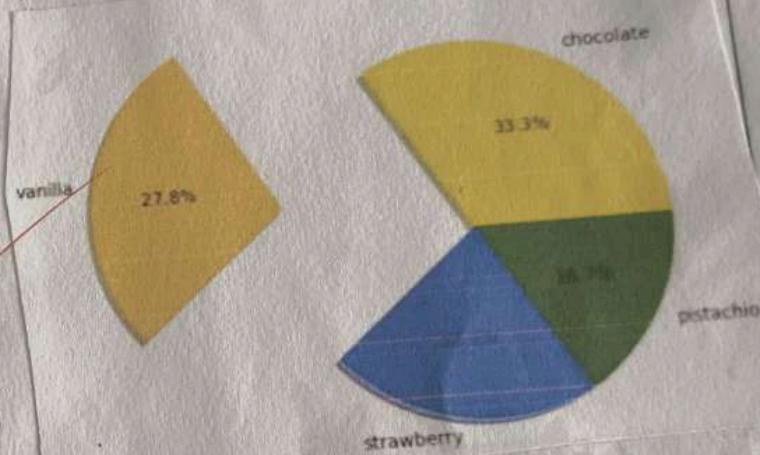


This graph represents  
the data of numbers  
in area

out put of area plot

This graph  
represents flavours  
of ice cream

out put of pie-chart  
flavours of icecream



### ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	09	
Planning and Execution/ Practical Simulation/ Programming	10	10	
Result and Interpretation	10	09	
Record of Applied and Action Learning	10	09	
Viva	10	09	
<b>Total</b>	<b>50</b>	<b>46</b>	

Signature of the Student: S. Yugandhar

Name: S. Yugandhar

Regn. No.: 241801350010

Page No..

\*As applicable according to the experim  
Two sheets per experiment (10-20) to

Signature of the Faculty:



School: SOET Campus: Vizianagaram  
Academic Year: 2024-25 Subject Name: DAVP Subject Code: \_\_\_\_\_  
Semester: I Program: B.TECH Branch: CSE Specialization: CN

Date: \_\_\_\_\_

## Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

### \* Coding Phase: Pseudo Code / Flow Chart / Algorithm

pseudo code to read csv files:

- 1) Improve Pandas library
- 2) Read the csv files into a data frame using pd.read\_csv
- 3) print the data frame

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used

\* Testing Phase: Compilation of Code (error detection)

```
# read csv file  
import pandas as pd  
df = pd.read_csv('yugandhar.csv')  
print("df")
```

S. No	Name	Marks
1	Kohli	100
2	Dhoni	80
3	KL Rahul	90
4	Paddikkal	80
5	Rohit	100

## ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	10	
Planning and Execution/ Practical Simulation/ Programming	10	10	
Result and Interpretation	10	10	
Record of Applied and Action Learning	10	09	
Viva	10	09	
<b>Total</b>	<b>50</b>	<b>48</b>	

Signature of the Faculty:

Signature of the Student: S.Yugandhar  
 Name: S.yugandhar  
 Regn. No.: 241801350010

Page No.....  
 \*As applicable according to the experimenter  
 Two sheets per experiment (10-20) to be used



School: .....

SDET

Campus: Vizianagram

Academic Year: 2024-25 Subject Name: D.A.V.P. Subject Code: .....

Semester: I Program: B.TECH Branch: CSE Specialization: CN

Date: .....

## Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

### Coding Phase: Pseudo Code / Flow Chart / Algorithm

pseudo code to read csv file:

- 1) import pandas as pd
- 2) Read the csv file into a data frame
- 3) print the data frame

pseudo code to visualize the csv file in scatter plot:

- 1) import Seaborn, matplotlib.pyplot & numpy libraries
- 2) Create a data frame with "data" columns "name" "marks"

3) Create a scatter plot with 'name' on x-axis & marks on y-axis

- 4) Display the scatter plot

pseudo code to visualize the csv file in box plot:

- 1) import Seaborn, matplotlib.pyplot and numpy libraries
- 2) Create a box plot with 'name' on x-axis and marks on y-axis
- 3) set the title of plot.
- 4) Display the plot.

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

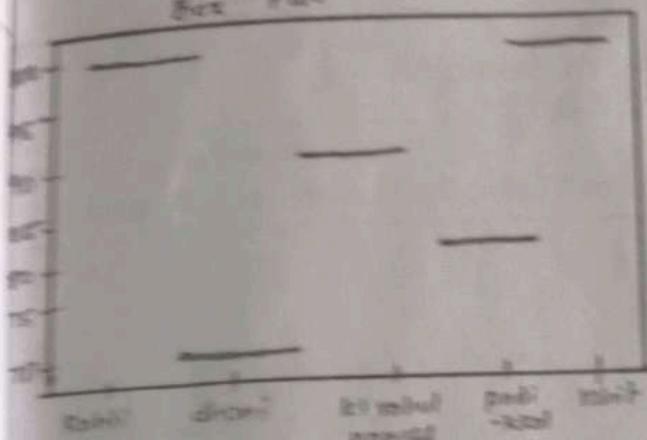
### \* Testing Phase: Compilation of Code (error detection)

```
# Read csv file
import pandas as pd
data = pd.read_csv("Yugandhar.csv")
print(data)

# To visualize the data in scatter plot
import Seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
sns.scatterplot(x='name', y='marks', hue="Sino",
                 data=data, alpha=1)
plt.show()

# To visualize data plot
import sea born as sns
import matplot.lib.pyplot as plt
import numpy as np
sns.boxplot(x="name", y="marks", data=data)
plt.title("box plot")
plt.show()
```

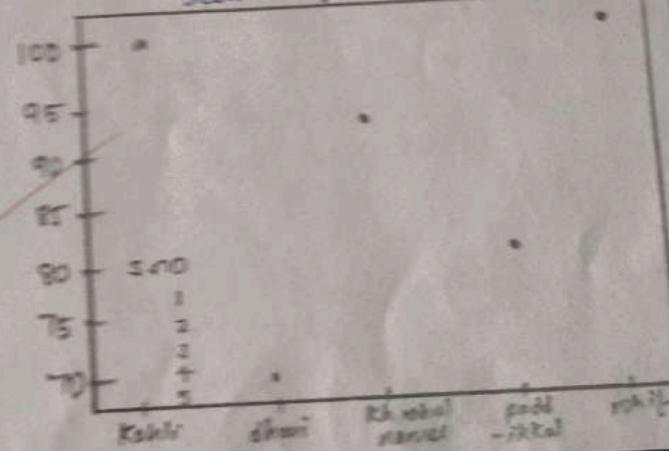
Box Plot



→ This CSV file represents  
the box plot of the  
given data.

→ This CSV file represents  
the scatter plot of  
the given data.

Scatter plot



### ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	10	
Planning and Execution/ Practical Simulation/ Programming	10	89	
Result and Interpretation	10	10	
Record of Applied and Action Learning	10	10	
Viva	50	49	
Total			

Signature of the Student: S. Yugandhar

Name: S. Yugandhar

Regn. No.: 241801350010

Page No. \_\_\_\_\_

Signature of the Faculty:

School: SOET Campus:  
 Academic Year: 2021-25 Subject Name: DAVP Subject Code:  
 Semester: 1 Program: BE-TECH Branch: CSE Specialization: CE  
 Date:

## Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

### \* Coding Phase: Pseudo Code / Flow Chart / Algorithm

#### Pseudo Code for Line plot

- 1) start
- 2) import the matplotlib library
- 3) Define the data points for x & y
- 4) Create a line plot with the data points
- 5) label the x-axis & y-axis
- 6) Add line plot title
- 7) Display the plot
- 8) End

#### Pseudo Code for Bar plot

- 1) start
- 2) Import the matplotlib library
- 3) Define the list of categories
- 4) Define the list of values color
- 5) Corresponding to each category.
- 6) Create bar plot with categories on the x-axis & values of y-axis
- 7) Set the title of the plot to Bar plot
- 8) Label the x-axis as Domain
- 9) Label the y-axis as Values
- 10) Display the plot
- 11) End

#### Pseudo Code for List plot

- 1) Start
- 2) import matplot lib
- 3) define data list
- 4) Create a Listogram with color and edge color
- 5) Set title as "List plot"
- 6) Label axis
- 7) Show plot
- 8) End.

#### Pseudo Code for Scatter plot

- 1) Start
- 2) import mat plot lib
- 3) Define x and y data
- 4) Create Scatter plot with color and marker
- 5) Set title and labels
- 6) Show plot
- 7) End

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

### \* Testing Phase: Compilation of Code (error detection)

#### Compilation of line plot:

```
import matplotlib.pyplot as plt
x = [1, 2, 3, 4, 5]
y = [1, 4, 9, 16, 25]
# Create a line plot
plt.plot(x, y)
plt.xlabel("x-value")
plt.ylabel("y-value")
plt.title("line")
plt.show()
```

#### Compilation of Hist plot:

```
import matplotlib.pyplot as plt
data = [1, 1, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4, 4, 4,
        5, 5, 5, 5, 5, 5]
plt.hist(data, color="skyblue",
         edgecolor="black")
plt.title("Hist plot")
```

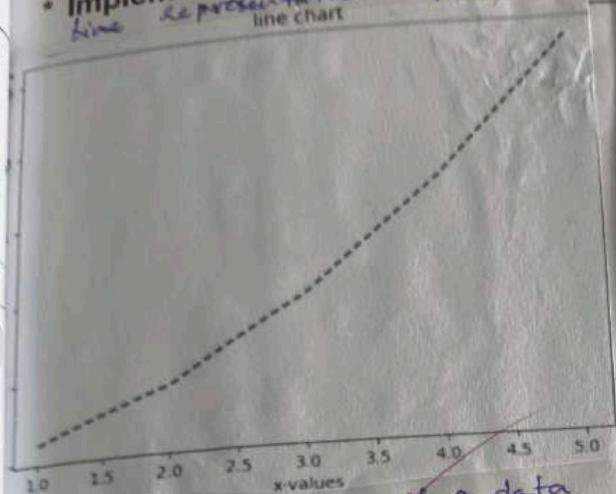
#### Compilation of Bar plot:

```
import matplotlib.pyplot as plt
categories = ["cn", "ic", "st", "Bca"]
values = [0.5, 1, 1, 0.5, 0]
plt.bar(categories, values)
plt.title("Bar plot")
plt.xlabel("Domain")
plt.ylabel("values")
plt.show()
```

#### Compilation of Scatter plot:

```
import matplotlib.pyplot as plt
x = [1, 2, 3, 4, 5]
y = [2, 4, 6, 8, 10]
plt.scatter(x, y, color="skyblue",
            marker="o")
```

\* Implementation Phase: Final Output (no error)

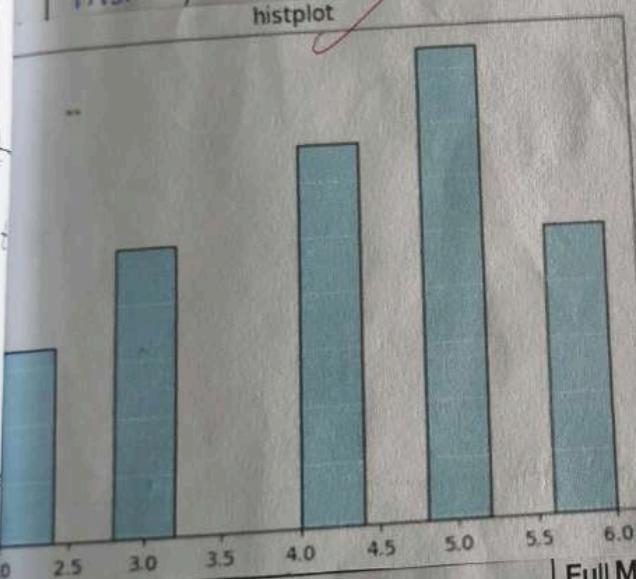


Line representation of a data  
line chart

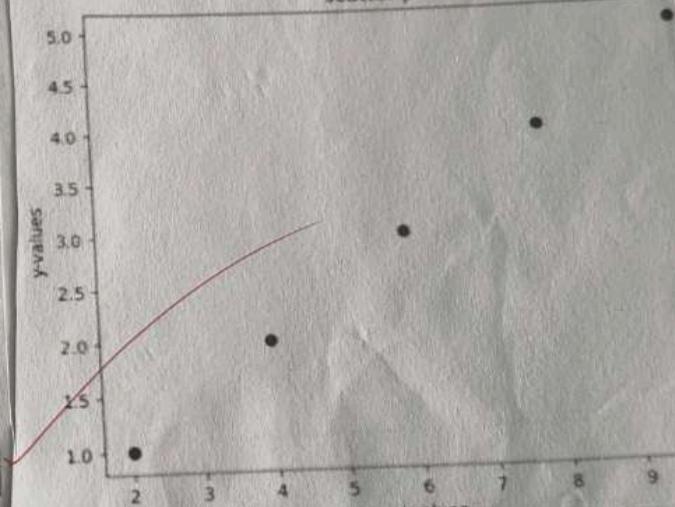
Applied and Action Learning  
Bar representation of a data



Bar representation of a data  
bar plot



Hist representation of a data  
histplot



Dot representation of a data  
scatter plot

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	10	
Planning and Execution/ Practical Simulation/ Programming	10	09	
Result and Interpretation	10	09	
Record of Applied and Action Learning	10	10	
Viva	10	10	
Total	50	48	

Signature of the Student: S. Yugandher

Name: S. Yugandher \*As applicable according to the experiment.  
Two sheets per experiment. Page No. be used.

Regn. No.: 241801350010

Signature of the Faculty



School: SO&T Campus: Vizianagaram  
Academic Year: 2024-25 Subject Name: D.A.V.D. Subject Code: \_\_\_\_\_  
Semester: I Program: E-TECH Branch: CSE Specialization: CSE  
Date: \_\_\_\_\_

## Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

Coding Phase: Pseudo Code / Flow Chart / Algorithm

1. start
2. install dash
3. import dash - components as dcc
4. import dash - html → components as html.
5. import numpy as np
6. import pandas as pd
7. import plotly express as px.
8. end.

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

## \* Testing Phase: Compilation of Code (error detection)

Impleme

### 1. Text styles

style = { "text-align": "center", "color": "#4CAF5B" }

### 2. Dropdown style

style = { "width": "45%", "display": "inline-block",  
"margin-right": "5px" }

### 3. Styling

style = { "border": "2px solid #4CAF50", "border-radius": "5px", "margin": "5px", "padding": "10px" }

### 4. Background

"Background color": "fffff ffd"

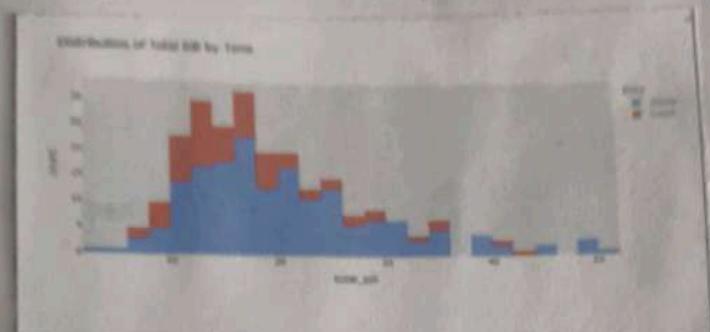
### 5. Box shadowing

"Box shadowing": "0 4px 8px rgba

0, 0, 0, 0

Conc  
Plann  
Pract  
Resu  
Reco  
Viva  
Total

## Implementation Phase: Final Output (no error)



ASSESSMENT			
Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	9	
Planning and Execution/	10	9	
Practical Simulation/ Programming	10	9	
Result and Interpretation	10	9	
Record of Applied and Action Learning	10	10	
Viva	50	46	
Total			

Signature of the Student:

Name :

Regn. No. :

\*As applicable according to the experiment  
Two sheets per experiment

School: SDET Campus: Vizianagaram  
Academic Year: 2024-25 Subject Name: DAVP Subject Code:  
Semester: 1 Program: B.TECH Branch: CSE Specialization: CSE  
Date:

Name of the Experiment:

Coding Phase: Pseudo Code / Flow Chart / Algorithm

- 1) start
2. pip install dash
3. import dash
4. from dash import dcc
5. from dash import html
6. from dash import input, output
7. import plotly.express as px.
8. import seaborn as sns.

Page No.....

\*As applicable according to the experiment  
Two sheets per experiment (10-20) to be used

## \* Testing Phase: Compilation of Code (error detection)

Impl

1. Search functionality

dcc. dropdown (

id = 'x-axis'; # Unique identifier for  
call b;

options = [ { 'label': col, 'value': val } for col in  
tips.columns ]

Value = 'total\_bill'; # Default

Style = { "width": "45%", "display":  
"inline-block",  
"margin-right": "5px" }

html.div [c

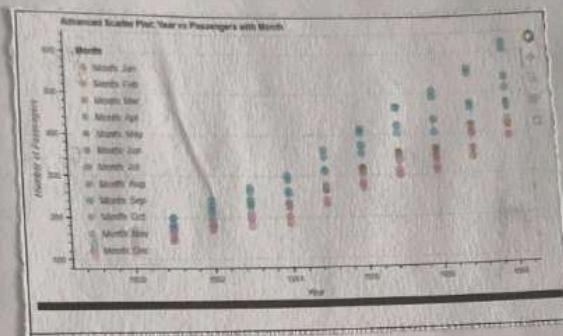
html.label ("Select year (if applicable):", style:  
{ "font-weight":  
"bold" },

dcc. dropdown (

id = 'filter-year'

options = [ { 'label': year, 'value': year } ]

## Implementation Phase: Final Output (no error)



## ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	9	
Planning and Execution/ Practical Simulation/ Programming	10	9	
Result and Interpretation	10	9	
Record of Applied and Action Learning	10	10	
Viva	50	47	
<b>Total</b>			

Signature of the Student:

Name :

Regn. No. :

\*As applicable according to the experiment.  
Two sheets per experiment. Page No. \_\_\_\_\_ be used.

Signature of the Faculty



School: SDET Campus: Muzicangara  
Academic Year: 2024-25 Subject Name: DAVP Subject Code: \_\_\_\_\_  
Semester: I Program: B.TECH Branch: ECE Specialization: CN  
Date: \_\_\_\_\_

Name of the Experiment:

### Coding Phase: Pseudo Code / Flow Chart / Algorithm

```
1. start
2. pip install dash board (dash)
3. import pandas as pd
4. import numpy as np
5. import plotly.express as px
6. import seaborn as sns
7. from dash import Dash, html, dcc.
8. app = Dash (- name -)
9. from dash import dash_table
10. from dash import Dash, dcc, html, input, output
11. END.
```

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

### \* Testing Phase: Compilation of Code (error detection)

```

# load dataset
df = sns.load_dataset("tips")
app = Dash(__name__)

# layout
app.layout = html.Div([
    html.H1("Tips dataset dashboard"),
    style={"text-align": "center"}
])

# Dropdown to select column
html.Label("Select x-axis for visualization"),
dcc.dropdown(
    id='x-axis',
    options=[{"label": col, "value": col}
]

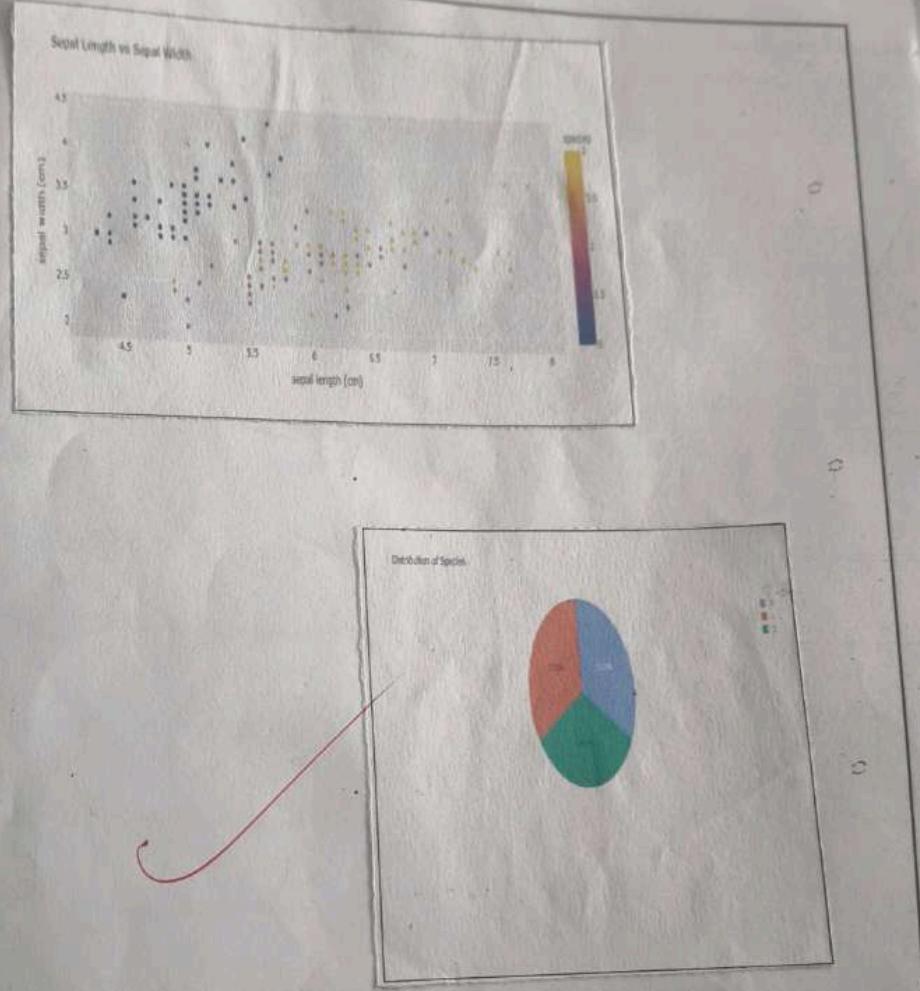
# pie chart
html.Label("Select column for pie chart")
dcc.dropdown(
    id='pie-chart-col',
    value='sex',
    options=[{"label": col, "value": col}
]

# bar chart
dcc.graph(id='pie-chart')

output('bar-chart').figure
Input('x-axis', Value)
def update_pie(column):
    return px.pie(title="Invalid statement")

```

## Implementation Phase: Final Output (no error)



## ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	9	
Planning and Execution/ Practical Simulation/ Programming	10	9	
Result and Interpretation	10	9	
Record of Applied and Action Learning	10	10	
Viva	10	10	
Total	50	47	

Signature of the Student:

Name :

Regn. No. :

\*As applicable according to the experiment.  
Two sheets per experiment. Program Note be used.

Signature of the Faculty



School: SOET Campus: Vigyanagaram  
Academic Year: 2024-25 Subject Name: D.A.V.P. Subject Code: \_\_\_\_\_  
Semester: I Program: B.TECH Branch: CSE Specialization: CN  
Date: \_\_\_\_\_

### Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

### Coding Phase: Pseudo Code / Flow Chart / Algorithm

1. start
2. from bokeh . plotting import figure
3. from bokeh . embed import title - html
4. from bokeh import CDN
5. from dash import Dash
6. from dash import html, dcc.
7. from dash import input, output
8. import pandas as pd
9. Import seaborn as sns
10. End.

Page No.....

\*As applicable according to the experiment  
Two sheets per experiment (10-20) to be used

## \* Testing Phase: Compilation of Code (error detection)

```

layout,
app.layout = html.Div([
    html.H2("Dash with Bokeh Visualiz",
    style = {"text-align": "center"})
    value = "Total bill",
    style = {"width": "50%", "margin": "auto"}
])

a. app call back
    Output('bokeh plot', 'children')
    Input('bokeh feature', 'value')
]

p = figure(
    title = "Bokeh scatter and line plot for",
    x_axis_label = 'Index',
    Input('bokeh - feature', 'value')
    p. circle(x = tips.index, y = tips.dl(feature), size =
        color = 'blue', alpha = 0.6, legend = 'label',
        'circle'
    )
    html_content = file.html(p, conn, "Bokeh plot"
        Conn
        Pract
        Resu
        Rec
        Viva
        Total
    )
)

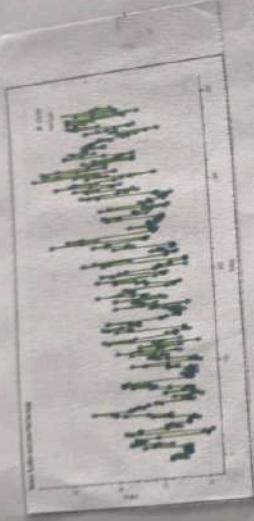
```

Turn in app

if .Name == "main":  
 app in ran + server (party = devlang\_true)

Applied and Action Learning

Implementation Phase: Final Output (no error)



ASSESSMENT		Marks Obtained	Remarks
Rubrics	Full Mark		
Concept	10		
Planning and Execution/ Programming	10		
Practical Simulation/ Programming	10		
Result and Interpretation	10		
Record of Applied and Action Learning	10		
Viva	50		
Total			

\* As applicable according to experiment  
Two sheets per experiment  
[Page] No. used

Signature of the Student:

Name : \_\_\_\_\_  
Regn. No. : \_\_\_\_\_

Signature of the Faculty

School: S.D.E.T Campus: Vizianagaram  
Academic Year: 2024 - 25 Subject Name: DDP Subject Code:  
Semester: I Program: B.Tech Branch: CSE Specialization: CSE

### Applied and Action Learning

(Learning by Doing and Discovery)

of the Experiment:

### Working Phase: Pseudo Code / Flow Chart / Algorithm

```
start
1. import plotting.figure, Show term bokeh, plotting
2. import output - notebook from bokeh.io
3. import Colourn Data Source from bokeh.model
4. import seaborn As sns
5. import pandas as pd
6.
7. import plotly Express as px
8. end.
```

Page No.....

\* As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

## \* Testing Phase: Compilation of Code (error detection)

After loading data set tips

Sex - color = ("Male": "blue", "female": "red")

Smoke - shape = ("yes": "triangle", "NO": "circle")

P: figure (title = "Advanced scatter plot: total bill vs tip

With sex and smokes", x-axis-label = "total bills", y-axis-label = "Tip (%)")

y-axis-label = "Tip (%)")

Smokes, shape in smokes - shape items()

p.scatter (

subset ["total\_bill"]

subset ("tips")

marker = shape

size = 15,

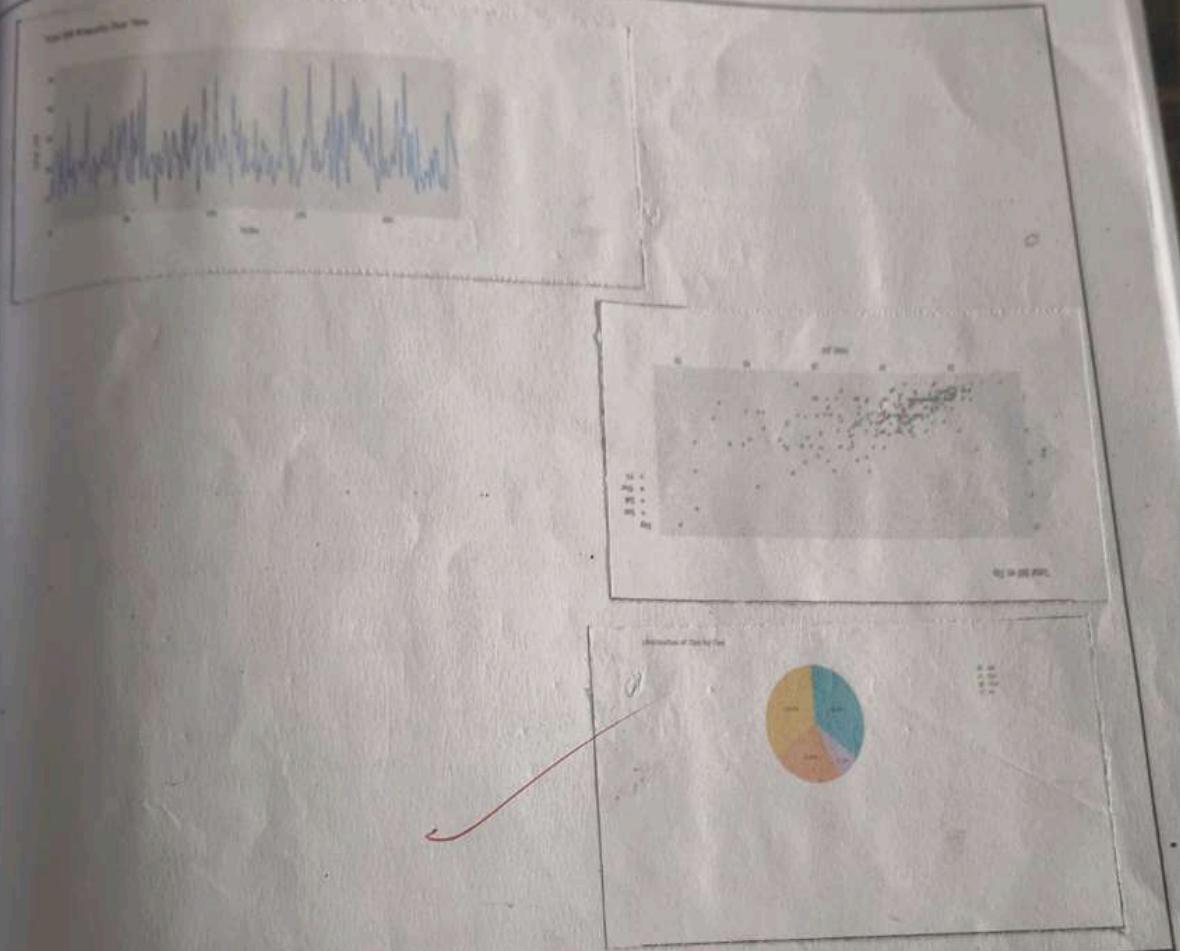
p.legend location = "top-left"

p.legend title = "legend"

show (p)

## Implementation Phase: Final Output (no error)

Applied and Action Learning



### ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	9	
Planning and Execution/ Practical Simulation/ Programming	10	9	
Result and Interpretation	10	9	
Record of Applied and Action Learning	10	10	
Viva	50	46	
Total			

Signature of the Student:

Name :

\*As applicable according to the experiment.  
Two sheets per experiment. Program Note be used.

Regn. No. :

Signature of the Faculty

periment  
to be used

School: S.D.E.T Campus: U.G. GianAngan  
Academic Year: 2024 - 25 Subject Name: D.A.V.P. Subject Code:  
Semester: I Program: B.TECH Branch: C.S.E Specialization: C.N.  
Date:

## Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

Coding Phase: Pseudo Code / Flow Chart / Algorithm

Pseudo code:

1. import libraries of beautiful Soup
2. # fetch web pages content;
  - > Define URI & send GET request
3. # parse HTML content
  - > parse content with beautiful soup
4. extract Data:
  - > find & print movie title
  - > find & print movie ratings.
5. print URL
  - > print the URL

Page No.....

\*As applicable according to the experiment.  
Two sheets per experiment (10-20) to be used.

## \* Testing Phase: Compilation of Code (error detection)

```
import requests:
```

```
from bs4 import BeautifulSoup
```

```
# fetch the web page content
```

```
URL = "https://www.imdb.com/chart/Top"
```

```
response = requests.get(URL)
```

```
# parse the HTML content
```

```
Soup = BeautifulSoup(response.content, "html")
```

```
# extract data
```

```
titles = Soup.find_all("td", class="title")  
for title in titles:  
    print(title.a.text)
```

```
# extract all the movie ratings
```

```
ratings = Soup.find_all("td", class="rating")
```

```
print(rating.strong.text)
```

```
print(URL)
```

## ASSESSMENT

Applied and Action Learning

Experiment	Full Mark	Marks Obtained
Experiment - 1	50	
Experiment - 2	50	
Experiment - 3	50	
Experiment - 4	50	
Experiment - 5	50	
Experiment - 6	50	
Experiment - 7	50	
Experiment - 8	50	
Experiment - 9	50	
Experiment - 10	50	
Experiment - 11	50	

Experiment	Full Mark	Marks Obtained
Experiment - 12	50	
Experiment - 13	50	
Experiment - 14	50	
Experiment - 15	50	
Experiment - 16	50	
Experiment - 17	50	
Experiment - 18	50	
Experiment - 19	50	
Experiment - 20	50	
Average Total	50	

Date: \_\_\_\_\_

*Signature of the Faculty*

### \* LEARNING OUTCOMES:

How the Applied and Action Learning encourages Critical Thinking, Problem Solving, Idea Generation and Skill Development etc.?

- \* Able to gain knowledge on Visualization with good story line and perform job of a data analysis.
- \* Able to analyze and visualize the data set
- \* Ability to design dashboard.

How the Applied and Action Learning encourages Leadership, Team Work, Reflection and Decision Making Capability etc.?

Leadership: hands-on-experience - learning provides opportunities to lead real-world projects  
 Decision making.  
 Collaborative problem solving.

*Signature of the Student*

Page No.....

\* One sheet per learning record to be used

Date: \_\_\_\_\_

## COURSE OUTCOMES (COs) ATTAINMENT

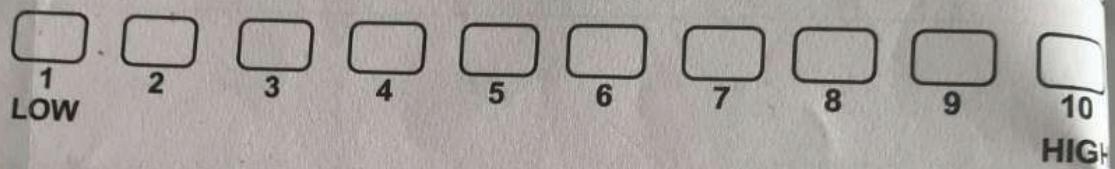
➤ Expected Course Outcomes (COs):

(Refer to COs Statement in the Syllabus)

- Perform effective data analysis and classification.
- Create effective data visualizations using libraries.
- Communicate data insights effectively.

➤ Course Outcomes (COs) Attained:

How would you rate your learning of the subject based on the specified COs?



➤ Learning Gap (If any):

Basic Syntax and Data Structure Syntax, Control flow, Data Structure (lists, dictionaries etc.), Data manipulation and Analysis.

➤ Books/Manuals Referred:

Date: \_\_\_\_\_

➤ Suggestions / Recommendations:  
(by the Course Faculty)

Signature of the Student

Date: \_\_\_\_\_

Signature of the Faculty

Page No. \_\_\_\_\_

\*One sheet per learning record to be submitted