



School: SOFT Campus: VZN

Academic Year: 24-15 Subject Name: DAVP Subject Code: 123

Semester: 1st Program: B.Tech Branch: CSE Specialization: CN

Date: 10/10/2024

Applied and Action Learning (Learning by Doing and Discovery)

of the Experiment:

Reading Phase: Pseudo Code / Flow Chart / Algorithm

Pseudo Code for line plot

- 1) Start
- 2) Import the matplotlib lib
- 3) Define data list
- 4) Create a line plot which 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 matplotlib lib
- 3) Define list of categories
- 4) Define values
- 5) Create barplot with x & y axis
- 6) Set the title
- 7) Label "x-axis" & "y-axis".
- 8) Display the plot
- 9) End.

→ pseudo code for scatter plot

- (1) Start
- (2) import matplotlib lib
- (3) Define x and y data
- (4) Create x and y data
- (5) Set title and labels
- (6) Show plot
- (7) End

→ pseudo code for histogram

- (1) Start
- (2) Import matplotlib lib
- (3) Define list of categories
- (4) Create histogram with color and Edge color
- (5) Set title as "hist plot"
- (6) Label axis
- (7) Show plot
- (8) End

Page No....!

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



Scanned with OKEN Scanner

* Testing Phase: Compilation of -

→ compilation code for line plot

```

x = [30, 50, 70, 90, 110]
y = [50, 80, 120, 150, 180]
plt.scatter(x,y, color="magenta", marker="*")

```

import matplotlib.pyplot as plt

a = [1, 2, 3, 8, 12]

b = [2, 4, 6, 8, 10]

create line plot

```

plt.plot(x,y)
plt.xlabel('a-Values')
plt.ylabel('b-Values')
plt.show()

```

→ compilation code for bar plot

import matplotlib.pyplot as plt

categories = ['CSD', 'AIML', 'CN', 'BCA', 'CIC']

values = [4, 5, 6, 6.5, 7]

plt.bar(categories, values)

plt.title("bar plot")

plt.xlabel('domains')

plt.ylabel('Values')

plt.show()

→ compilation code for histogram

Histogram

import matplotlib.pyplot as plt

a = [12, 26, 38, 50, 62]

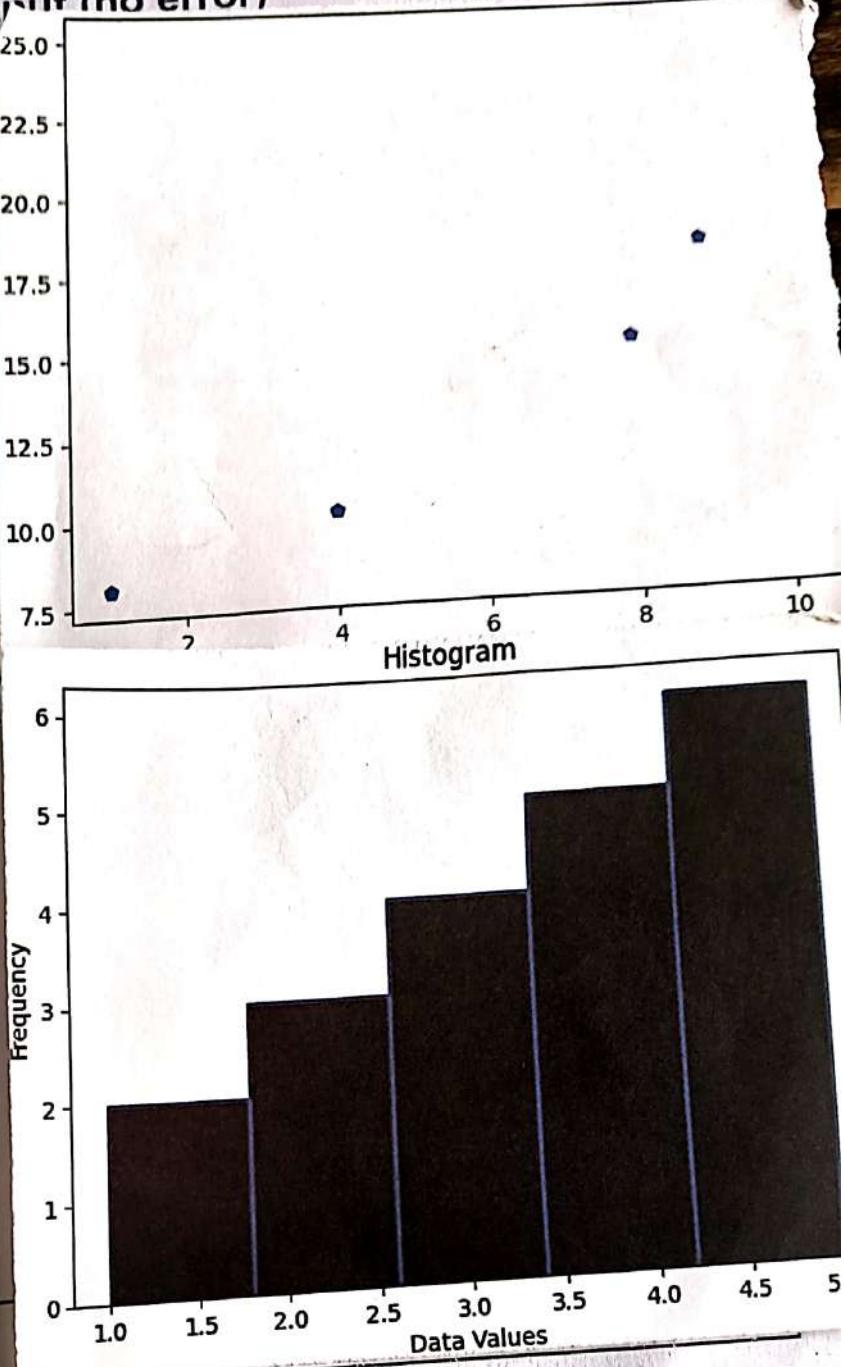
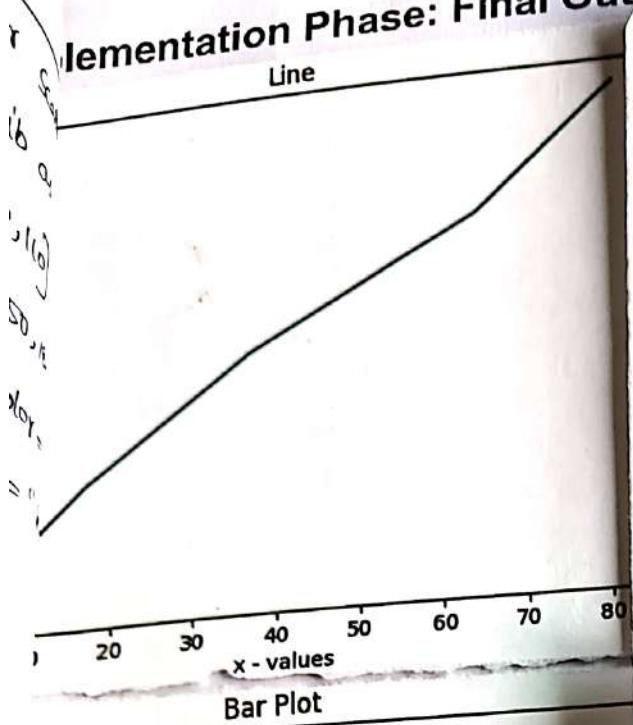
plt.hist(a, color="red", bins=5)

plt.title('histogram')

plt.xlabel('x-axis')

plt.show()

Implementation Phase: Final Output (no error)



ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Designing and Execution/	10	10	
ical Simulation/ Programming	10	9	
It and Interpretation	10	10	
ord of Applied and Action Learning	10	10	
	10	9	
	50	40	

Signature of the Student: M. Prem Sasi

Name: M. Prem Sasi *As applicable according to the experiment.
Sheets per experiment: 10 No. of user: 1

of the Faculty



Applied and Action Learning

(Learning by Doing and Discovery)

Goal of the Experiment :

Learning Phase: Pseudo Code / Flow Chart / Algorithm

→ Histogram

- (1) start
- (2) import matplotlib.pyplot
- (3) Define data list
- (4) create histogram (with bins, color, Edge color)
- (5) Set title as "Histogram".
- (6) Set x-label as "data values" and y-label as "frequency".
- (7) Show plot
- (8) End

→ Bar plot

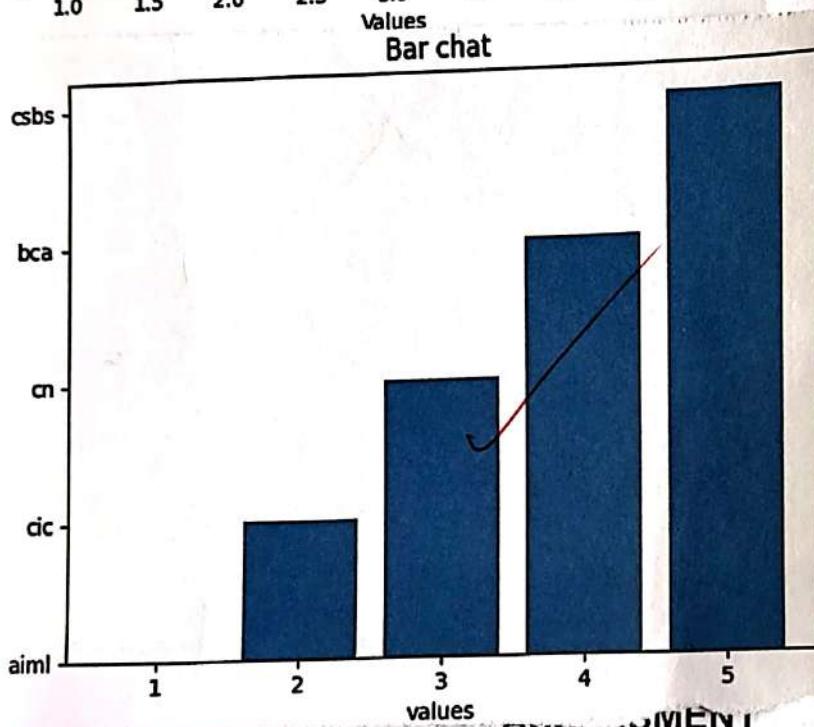
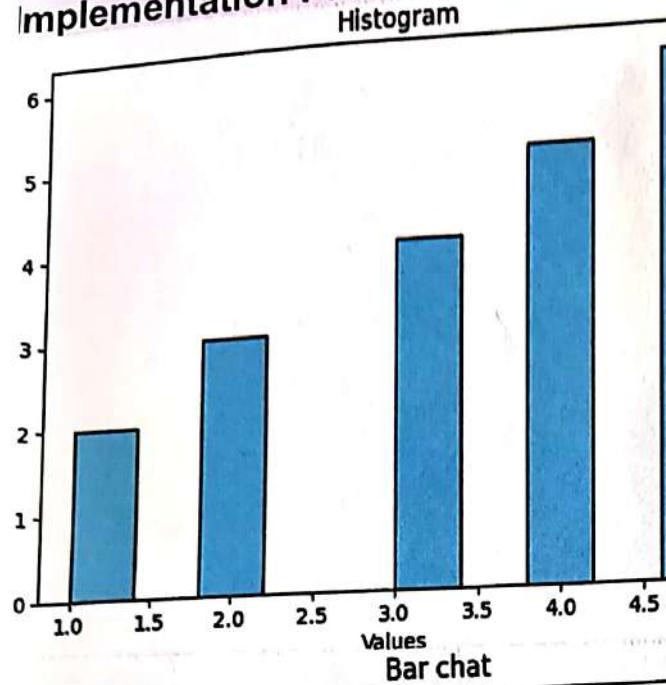
- (1) start
- (2) import matplotlib.pyplot
- (3) Define categories and values
- (4) Create a barplot (color & edge color)
- (5) Set title as "Bar plot"
- (6) Set x-label as "Domain" and y-label as "Values".
- (7) Show plot
- (8) End

* Testing Phase: Compilation of Code

```
import matplotlib.pyplot as plt  
data = [22, 45, 60, 70, 90]  
plt.hist(data, color = "green", bins=8)  
plt.title("histogram")  
plt.xlabel("x-axis")  
plt.show()
```

```
import matplotlib.pyplot as plt  
Categories = ['CSD', 'CN', 'CIC', 'BCA', 'AIML']  
Values = [25, 35, 45, 55, 65]  
plt.bar(Categories, Values, color = "Black")  
plt.title("Bar plot")  
plt.xlabel("Domains")  
plt.ylabel("Values")  
plt.show()
```

Implementation Phase: Final Output (no error)



Rubrics	Full Mark	Marks Obtained	Remarks
concept	10	10	
Planning and Execution/ Practical Simulation/ Programming	10	9	
Result and Interpretation	10	10	
Record of Applied and Action Learning	10	10	
Final Report	10	9	
Total	50	48	

Signature of the Faculty

Signature of the Student:

Name : M. Prem Saini *As applicable according to the experiment.
Regn. No. : 24180135006 Two sheets per experiment
Program No. be used.

Applied and Action Learning

(Learning by Doing and Discovery)

Goal of the Experiment:

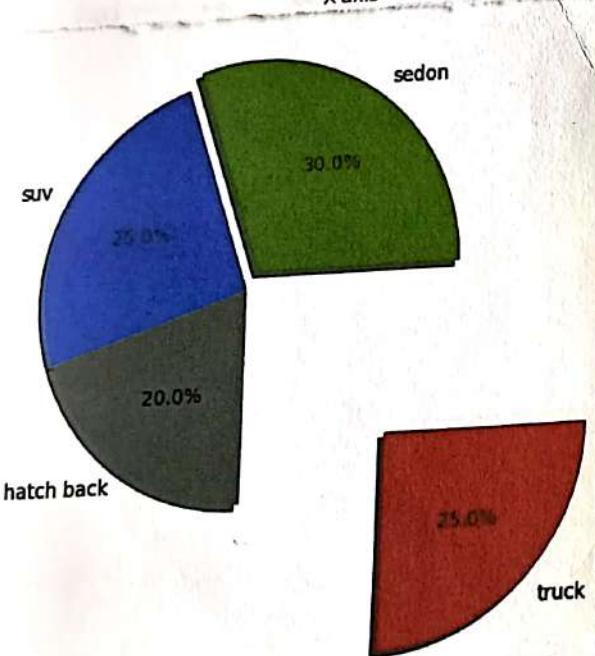
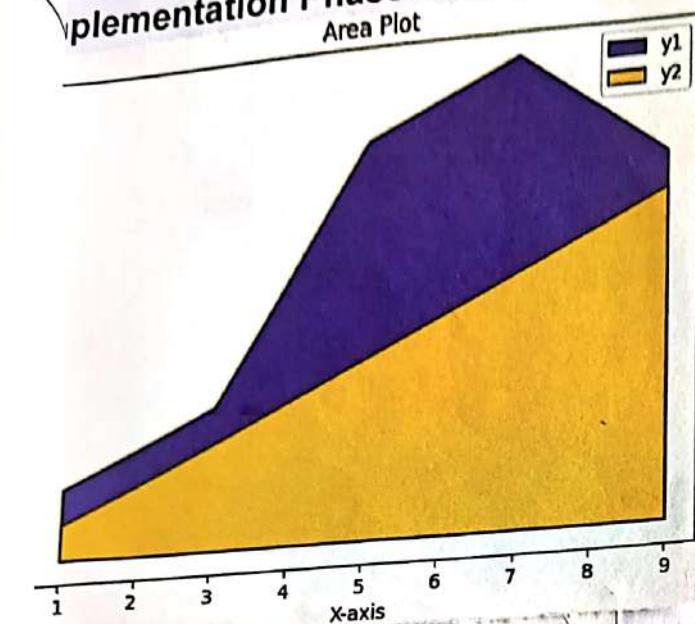
Coding Phase: Pseudo Code / Flow Chart / Algorithm

- 1) import matplotlib library
 - 2) create lists of x, y_1, y_2 for the data
 - 3) Create first area plot between x, y_1 with color and transparency
 - 4) Create second area plot between x, y_2 by colour and transparency.
 - 5) Set title for the plot.
 - 6) Label x-axis and y-axis
 - 7) Add a legend to differentiate between two areas
 - 8) Display the plot.
-
- 1) import matplotlib library.
 - 2) Create a list for label, sizes, colors and create piechart with { sizes, explode, labels, colors, percentage, shadow)
- Set the title for the plot
- Display the plot.

* Testing Phase: Compilation of Code (error detection)

```
# Create area plot
import matplotlib.pyplot as plt
x = (20, 24, 28, 32, 36)
y1 = (316, 9, 12, 15)
y2 = (218, 14, 18, 22)
plt.fill_between(x, y1, color = "blue", alpha = 1, label
                 = "y1", edgecolor = "black")
plt.fill_between(x, y2, color = "red", alpha = 1, label
                 = "y2", edgecolor = "black")
plt.title('Area plot')
plt.xlabel('x-axis')
plt.ylabel('y-axis')
plt.legend()
label = ["kiwi", "mango", "apple", "grapes"]
sizes = [35, 20, 30, 15]
colors = ['yellow', 'red', 'green', 'blue', 'violet']
explode = (0.1, 0, 0, 0)
plt.pie(sizes, explode = explode, labels = labels, colors =
        auto
plt = ' ', shadow = True)
plt.title("flavours of fruits")
plt.show()
```

Implementation Phase: Final Output (no error)



ASSESSMENT

Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	10	
Planning and Execution/ actical Simulation/ Programming	10	9	
Result and Interpretation	10	10	
Record of Applied and Action Learning	10	10	
Overall	10	9	
Total	50	48	

Signature of the Student:

Name: M Prem Sai

Regn. No.: 241801350008

Page No.....

nature of the Faculty:



School: SOET Campus: VZN
Academic Year: 2023-24 Subject Name: DAVP Subject Code: 1018
Semester: 1st Program: B.Tech Branch: CSE Specialization: CN

Date:

Applied and Action Learning (Learning by Doing and Discovery)

Code of the Experiment:

Doing Phase: Pseudo Code / Flow Chart / Algorithm

→ pseudo code for pie chart plot:-

- 1) Start
- 2) import matplotlib lib
- 3) Define categories and values
- 4) Define explode and colors
- 5) Create pie chart with (autopct=shadow, explode colors)
- 6) Set title as "pie plot".
- 7) Show plot
- 8) End

→ pseudo code for Bar plot :-

- 1) Start
- 2) import matplotlib .lib
- 3) Define categories list
- 4) Define values list
- 5) Create barplot with (color, edge color)
- 6) Set title as "Bar plot".
- 7) Label the axis
- 8) Show plot
- 9) End

Page No.....

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



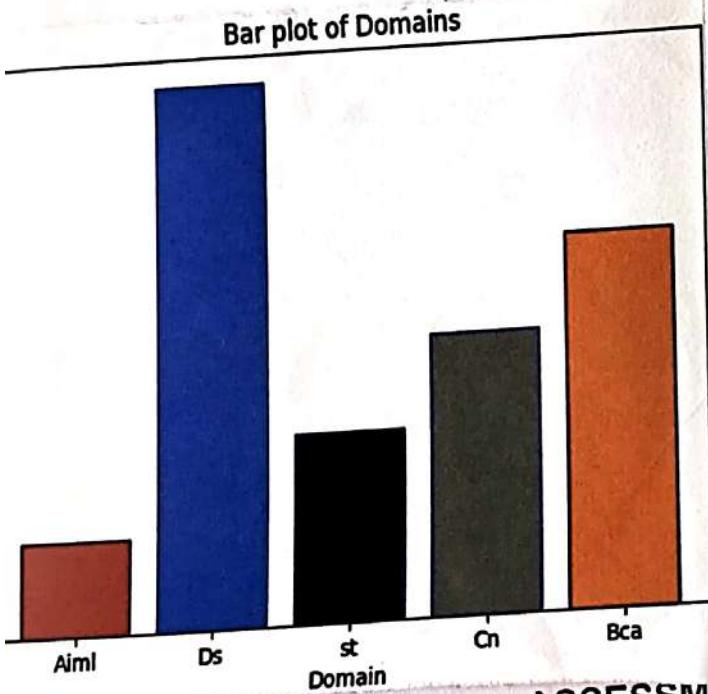
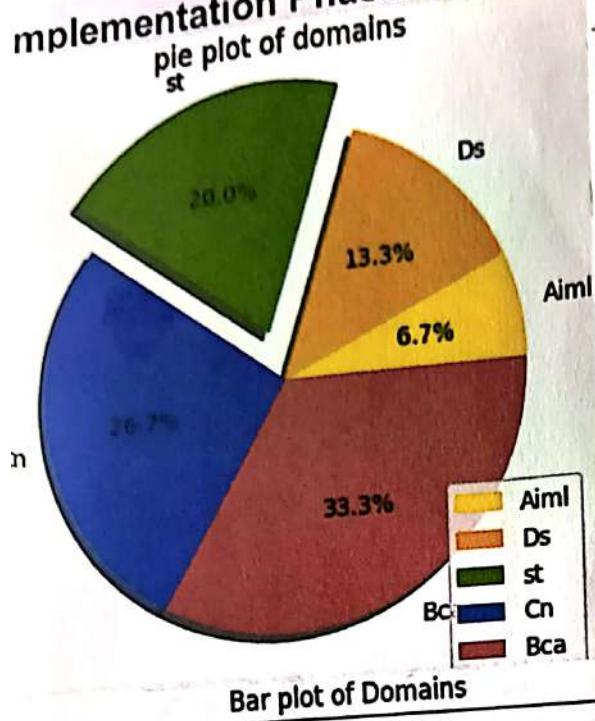
Scanned with OKEN Scanner

* Testing Phase: Compilation of Code (error detection)

```
import matplotlib.pyplot as plt  
Categories = ['CSO', 'CN', 'BCA', 'CJC', 'AIML']  
Values = [20, 20, 40, 50, 60]  
explode = [0, 0.1, 0, 0, 0]  
Colours = ["Green", "Blue", "Yellow", "light green", "orange"]  
plt.pie(Values, labels=Categories, autopct=%  
        shadow=True, explode=explode, colors=Colours)  
plt.title("Pie plot of domains")  
plt.show()
```

```
import matplotlib.pyplot as plt  
Categories = ['CSO', 'W', 'BCA', 'CJC', 'AIML']  
Values = [4.1, 2.5, 1.5, 3.5, 6.5]  
Color = ["blue", "orange", "grey", "violet", "green"]  
plt.bar(Categories, Values, color=Color, edgecolor="green")  
plt.title("Barplot of Domain")  
plt.xlabel("Domain")  
plt.ylabel("Values")  
plt.show()
```

Implementation Phase: Final Output (no error)



ASSESSMENT			
Rubrics	Full Mark	Marks Obtained	Remarks
Concept	10	10	
Planning and Execution/	10	10	
Practical Simulation/ Programming	10	10	
Result and Interpretation	10	9	
Record of Applied and Action Learning	10	10	
Viva	10	9	
Total	50	49	

Signature of the Faculty

Signature of the Student: M. Aman Saj

Name: M. Aman Saj *As applicable according to the experiment.
Two sheets per experiment. Please Note: No be used.



School: SOBT Campus: VBM
Academic Year: 2025 Subject Name: DAVP Subject Code: 10116
Semester: 1st Program: BTech Branch: CSE Specialization: CN
Date:

Applied and Action Learning

(Learning by Doing and Discovery)

Time of the Experiment:

Coding Phase: Pseudo Code / Flow Chart / Algorithm

Pseudo code for scatter plot

- 1) Import matplotlib lib.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 and marker
- 6) Display the plot.

Pseudo code for the plot.

- (1) Import matplotlib lib pyplot as plt
- (2) create a list of x and y
- (3) plot the data with specified markers, colors & 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 lib.pyplot as plt
- 2) create a list data
- 3) create a histogram with colors and edge colors
- 4) set the title of the plot
- 5) label the x-axis & y-axis
- 6) Display the plot

Pseudo Code for Bar chart

- (1) Import matplotlib lib pyplot as plt
- (2) create lists for Categories & Values
- (3) plot the bar chart with Values 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.



Scanned with OKEN Scanner

* Testing Phase: Compilation of Code (error detection)

Scatter plot

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

```
x = [1, 2, 3, 4, 5]
```

```
y = [2, 4, 6, 8, 10]
```

```
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, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5,  
5, 5]
```

~~```
plt.hist(data, color = "yellow")
```~~~~```
edge_color = "black")
```~~~~```
plt.title("Histogram")
```~~~~```
plt.xlabel("Values")
```~~

Line plot

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

```
X = [1, 2, 3, 4, 5]
```

```
y = [2, 4, 6, 8, 10]
```

```
plt.plot(X, Y, marker_size = 10,  
color = "black", line_style = "solid")
```

```
plt.xlabel("X-Values")
```

```
plt.ylabel("Y-Values")
```

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

```
plt.show()
```

Bar chart

```
Categories = ["AIML", "CIC", "CN",  
"CSO"]
```

```
Values = [1, 2, 3, 4, 5]
```

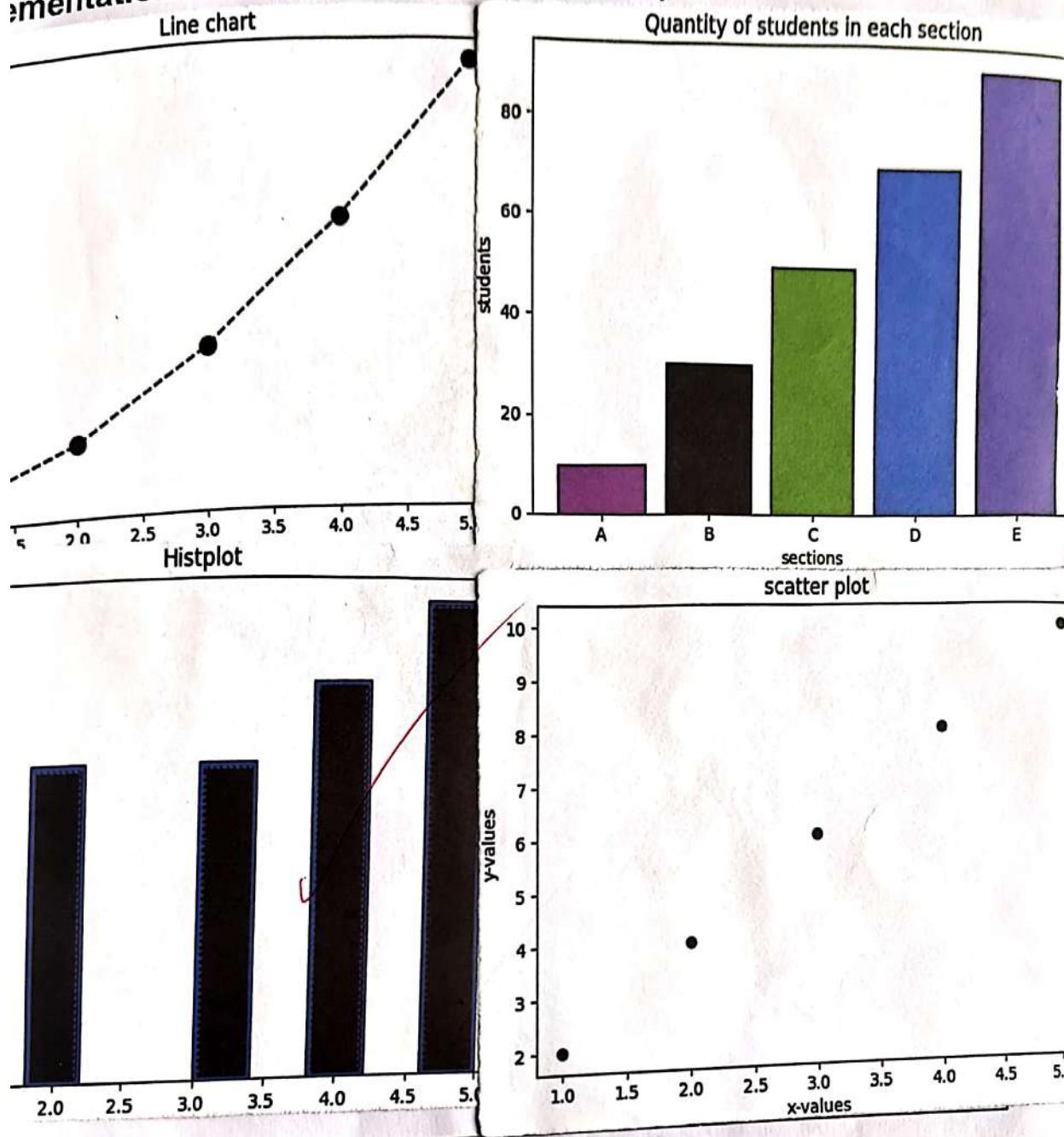
```
plt.bar(Values, Categories)
```

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

```
plt.xlabel("Values")
```

```
plt.ylabel("domains")
```

Implementation Phase: Final Output (no error)



ASSESSMENT

| Rubrics | Full Mark | Marks Obtained | Remarks |
|--------------------------------|-----------|----------------|---------|
| Implementation | 10 | 10 | |
| Execution/ | 10 | | |
| Simulation/ Programming | 10 | 10 | |
| Interpretation | 10 | 9 | |
| of Applied and Action Learning | 10 | 9 | |
| | 10 | 9 | |
| | 50 | 42 | |

(Signature of the Faculty)

Signature of the Student: *M. Premlal*

Name: *M. Premlal* *As applicable according to the experiment.
Two sheets per experiment Program No. be used.



School: SOAT Campus: V2M
Academic Year: 24-25 Subject Name: DAVP Subject Code: 1018
Semester: 1st Program: B.Tech Branch: CSE Specialization: AI
Date:

Applied and Action Learning

(Learning by Doing and Discovery)

of the Experiment :

Doing Phase: Pseudo Code / Flow Chart / Algorithm

→ pseudo code for bar plot :-

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

pseudo code for scatter plot :-

- (1) import matplotlib 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 label
- (7) Display chart

Page No.....

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



Scanned with OKEN Scanner

* Testing Phase.

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

```
category = ("A", "B", "C", "D", "E")
```

```
values = [10, 20, 40, 60, 80]
```

```
color = ("green", "blue", "purple", "red", "grey")
```

```
plt.bar(categories, values, color=color, edgecolor="white")
```

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

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

```
plt.title('Barplot')
```

```
plt.show()
```

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

```
x = ["A", "B", "C", "D", "E"]
```

```
y = [22, 33, 44, 55, 66]
```

```
plt.scatter(x, y, color="red", marker="R")
```

```
plt.title("Quantity of rooms in each floor")
```

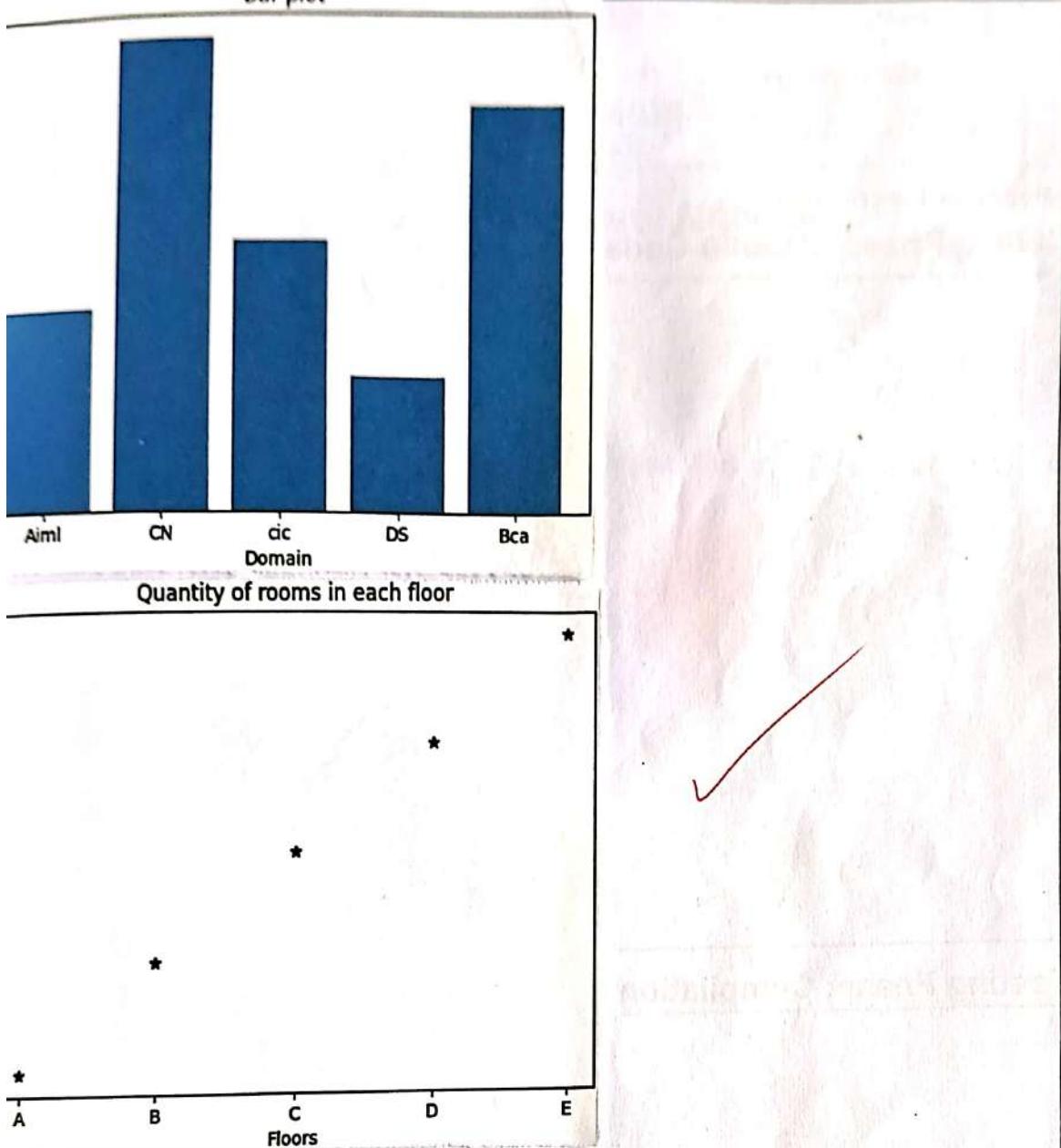
```
plt.xlabel("floors")
```

```
plt.ylabel("rooms")
```

```
plt.show()
```

Implementation Phase: Final Output (no error)

Bar plot



ASSESSMENT

| Rubrics | Full Mark | Marks Obtained | Remarks |
|---|-----------|----------------|---------|
| Conceptualization | 10 | 10 | |
| Planning and Execution/ Practical Simulation/ Programming | 10 | 9 | |
| Result and Interpretation | 10 | 10 | |
| Overall Applied and Action Learning | 10 | 10 | |
| Total | 50 | 45 | |

Signature of the Student: M Prem Sai

Name: M Prem Sai

Regn. No.: 241801350028

Page No.....

Signature of the Faculty:



School: SOFT Campus: VEN
Academic Year: 2021-25 Subject Name: DATA Subject Code: 101
Semester: 1ST Program: D.Tech Branch: CSE Specialization: AI
Date:

Applied and Action Learning

(Learning by Doing and Discovery)

of the Experiment :

ding Phase: Pseudo Code / Flow Chart / Algorithm

pseudo code to read csv files :-

- | Import pandas library
- | Read the csv file into a data frame using pd.read_csv
- | print the data frame

Page No.....

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



Scanned with OKEN Scanner

* Testing Phase: Compilation of Code (error detection)

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

Implementation Phase: Final Output (no error)

```
import pandas as pd
```

```
df = pd.read_csv('HarshitH.csv')
```

```
print(df)
```

| S.NO | Name | reg.no | Marks |
|------|-----------|--------|-------|
| 0 | Munna | 1 | 80 |
| 1 | HarshitH | 2 | 90 |
| 2 | Abhi | 3 | 65 |
| 3 | Kushwauth | 4 | 50 |
| 4 | Ruban | 5 | 40 |

ASSESSMENT

| Rubrics | Full Mark | Marks Obtained | Remarks |
|--|-----------|----------------|---------|
| Concept | 10 | 10 | |
| Planning and Execution/
Technical Simulation/ Programming | 10 | 9 | |
| Result and Interpretation | 10 | 10 | |
| Overall Applied and Action Learning | 10 | 10 | |
| Total | 50 | 48 | |

Signature of the Student: 

Name : M. Prem Lal *As applicable according to the experiment.
 Two sheets per experiment Page No. to be used.
 Regn. No. : 2413018006

Signature of the Faculty

School: WFT Campus: HDM
Academic Year: 2015-16 Subject Name: DOYP Subject Code: 1016
Semester: 1st Program: B.Tech Branch: CSE Specialization: AI
Date: _____

Applied and Action Learning (Learning by Doing and Discovery)

of the Experiment:

ding Phase: Pseudo Code / Flow Chart / Algorithm

pseudo code for 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:

- 1) import seaborn, matplotlib.pyplot & numpy libraries
- 2) create a data frame with "data" columns, "name", "marks" and "sno".
- 3) create a scatterplot 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 boxplot 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.



Scanned with OKEN Scanner

* Testing Phase: Compilation of Code (error detection)

Read csv file.

```
import pandas as pd
```

```
data = pd.read_csv("Harshith.csv")
```

```
print(data)
```

To visualise the data in scatter plot

```
import seaborn as sns
```

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

```
import numpy as np
```

```
sns.scatterplot(x="name", y="marks", hue="class",
```

```
data = data.alpha(1))
```

To visualise data plot

```
import seaborn as sns
```

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

```
import numpy as np
```

```
sns.boxplot(x="name", y="marks", data=data)
```

```
plt.title("Box plot")
```

```
plt.show()
```

Implementation Phase: Final Output (no error)

Q

L

ASSESSMENT

| Rubrics | Full Mark | Marks Obtained | Remarks |
|---|-----------|----------------|---------|
| Report | 10 | 10 | |
| Designing and Execution/
Numerical Simulation/ Programming | 10 | 9 | |
| Result and Interpretation | 10 | 9 | |
| Overall Applied and Action Learning | 10 | 9 | |
| | 10 | 9 | |
| | 50 | 46 | |

Signature of the Student: Mohan S

Name : M. Prem Saini *As applicable according to the experiment.
Regn. No. : 2416085 Two sheets per experiment. Program No. to be used.

Signature of the Faculty



| | | | |
|----------------|-----------------|-----------------|--------|
| School: | SOAT | Campus: | 1/2M |
| Academic Year: | 2015-16 | Subject Name: | DAV |
| Semester: | 1 st | Program: | B.Tech |
| Branch: | CSE | Specialization: | AI |
| Date: | | | |

Applied and Action Learning
(Learning by Doing and Discovery)

of the Experiment:

doing Phase: Pseudo Code / Flow Chart / Algorithm

pseudo code for bubble chart:-

- , import matplotlib lib pyplot
- , create lists of x,y and size for coordinates and bubbles
- , plot the bubble chart using the x,y and large lists
and specify the color and transparency
- , set the title for the plot
- , label the x-axis and y-axis
- , display the plot.

pseudo code for heat map:-

import the Seaborn matplotlib lib pyplot and numpy lib

create 10x10 matrix of random data using numpy

plot the heatmap with the 'coolwarm' k color

set the title of the plot

label x-axis and y-axis

show the plot

* Testing Phase: Compilation of Code (error detection)

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

```
x = [30, 40, 50, 60, 70] 6, 10, 15, 20, 25]
```

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

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

```
plt.scatter(x, y, s=size, color="black", alpha=1)
```

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

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

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

```
plt.show()
```

```
import Seaborn as sns
```

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

```
import numpy as np
```

```
data = np.random.rand(4,4)
```

```
sns.heatmap(data, (map="coolwarm", annot=True))
```

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

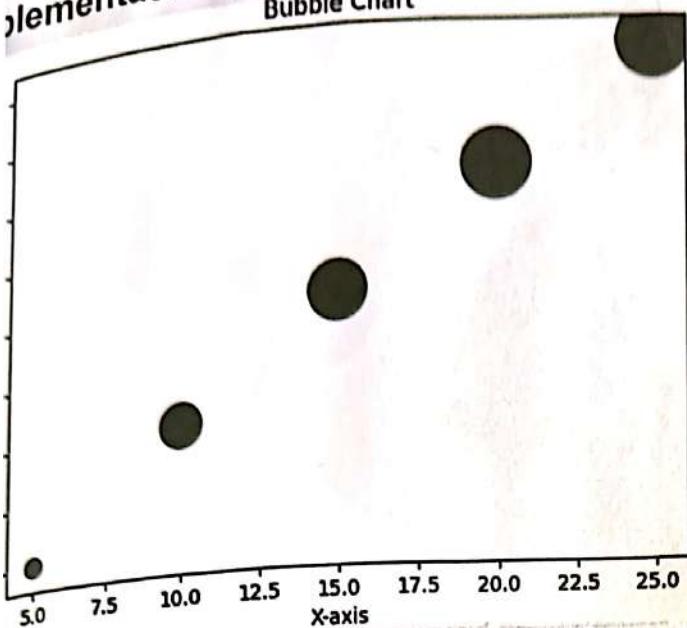
```
plt.xlabel("x-axis")
```

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

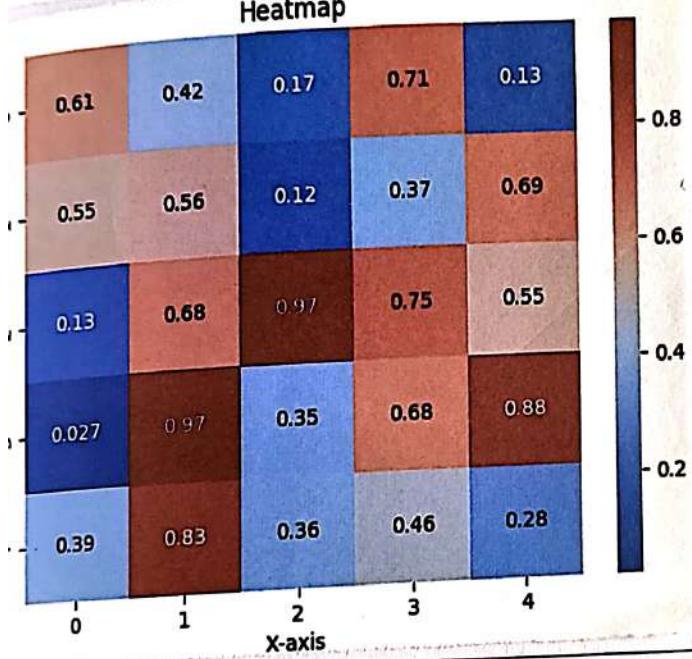
```
plt.show()
```

Implementation Phase: Final Output (no error)

Bubble Chart



Heatmap



ASSESSMENT

| Rubrics | Full Mark | Marks Obtained | Remarks |
|--|-----------|----------------|---------|
| Concept | 10 | 10 | |
| Planning and Execution/
Practical Simulation/ Programming | 10 | 9 | |
| Result and Interpretation | 10 | 9 | |
| Record of Applied and Action Learning | 10 | 10 | |
| Total | 50 | 49 | |

Signature of the Student: *[Signature]*

Name: Mr.Prem.Sai

Regn. No.: 241801356008

Page No.....

Signature of the Faculty:



School: SOFT Campus: VEN
Academic Year: 2021-22 Subject Name: M.V.P Subject Code: 1014
Semester: 11 Program: B.Tech Branch: CSE Specialization: CSE
Date:

Applied and Action Learning

(Learning by Doing and Discovery)

of the Experiment:

Planning Phase: Pseudo Code / Flow Chart / Algorithm

- 1) Start
- 2) pip install dashboard (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.



Scanned with OKEN Scanner

* 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}]

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

## bar plot
output('bar-plot', figure,
    input('x-axis', 'Value'),
    def "update - pie (column):
        return ex.pie(title = "Invalid statement.

# Run in app.
if __name__ == 'main':
    app.run_server(port = 8055, debug = True)
```

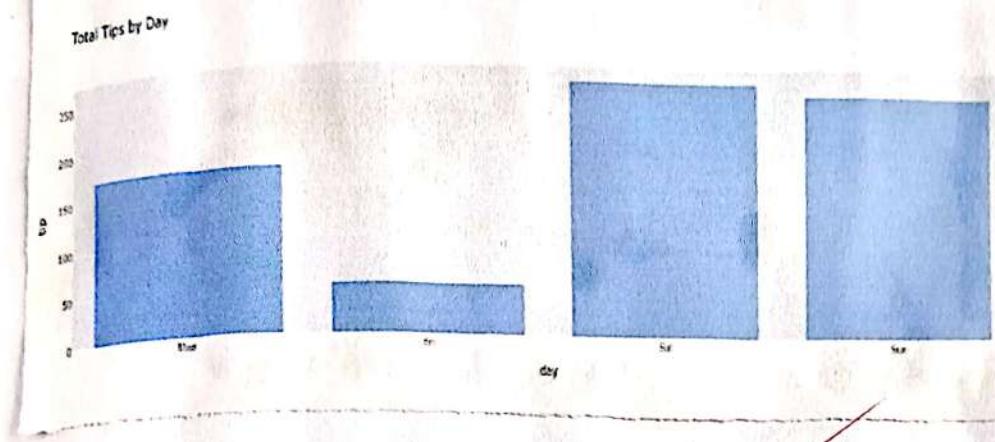
*As applicable according to the
Two sheets per experiment (10)



Scanned with OKEN Scanner

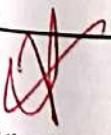
Implementation Phase: Final Output (no error)

Tips Dashboard



ASSESSMENT

| Rubrics | Full Mark | Marks Obtained | Remarks |
|---|-----------|----------------|---------|
| Experimentation | 10 | 10 | |
| Planning and Execution/ Technical Simulation/ Programming | 10 | 9 | |
| Data Interpretation | 10 | 8 | |
| Overall Applied and Action Learning | 10 | 10 | |
| | 10 | 10 | |
| | 50 | 47 | |


Signature of the Faculty

Signature of the Student: 

Name : M.PremSai *As applicable according to the experiment.
Regn. No. : 24180150006 Two sheets per experiment. Program Note be used.



School: S.O.T Campus: UZM
Academic Year: 2024-25 Subject Name: P.A.V.P Subject Code: I.A.B
Semester: 1st Program: B.Tech Branch: CSE Specialization: AI
Date:

Applied and Action Learning

(Learning by Doing and Discovery)

of the Experiment: Customise dashboard components, including layout & elements
during Phase: Pseudo Code / Flow Chart / Algorithm

Pseudo Code for customising dashboard components including layout and interactive elements, to enhance user experience tips
data.

- (1) Start
- (2) Import dash, dec, html, plotly_express, Seaborn, pandas
- (3) Load the dataset
- (4) Create the dash.app
- (5) Define app layout
- (6) Define the call back function
- (7) Run the app
- (8) END.

Page No.....

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



Scanned with OKEN Scanner

* Testing Phase: Compilation of Code (error detection)

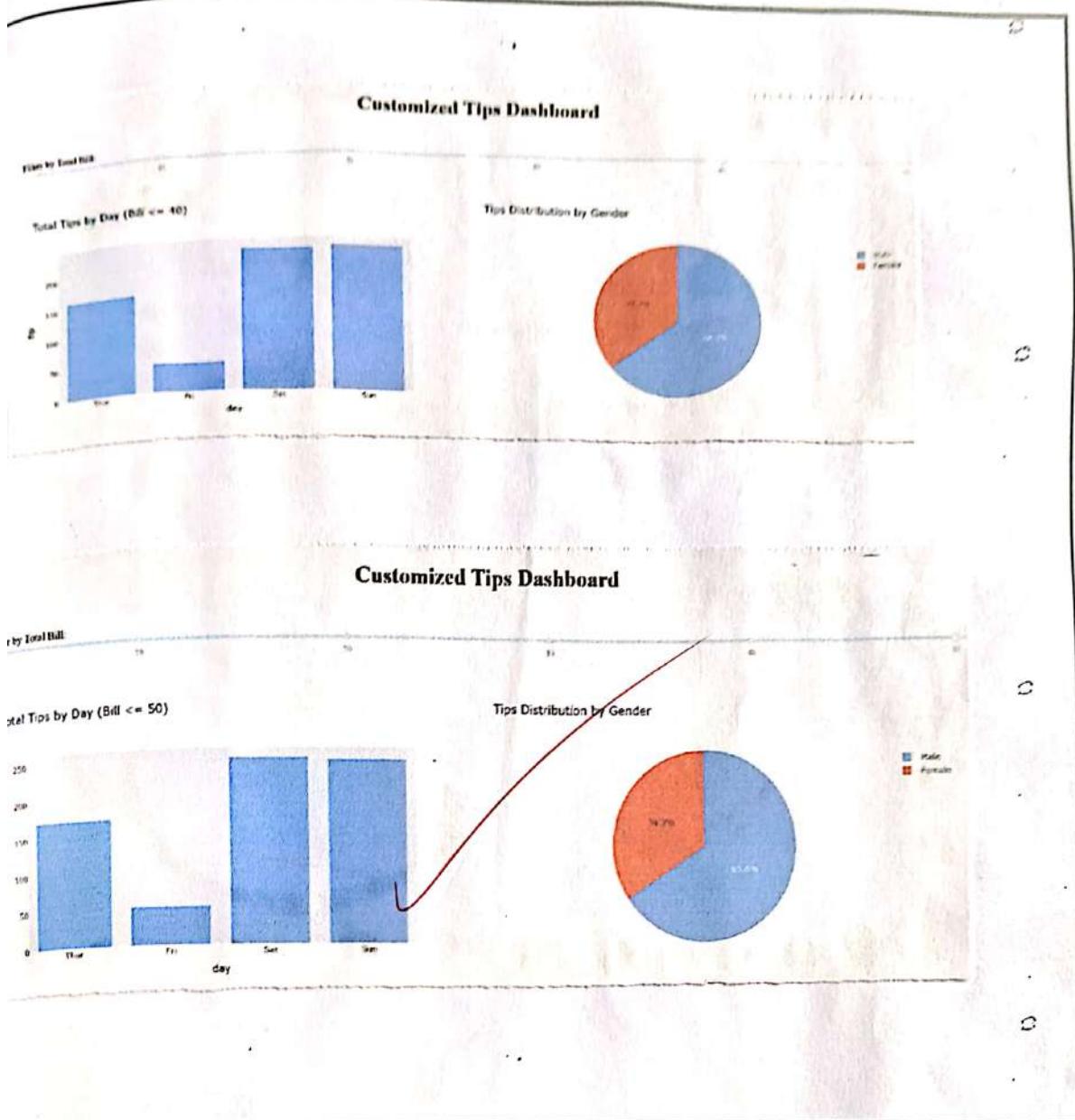
```

import dash
from dash import dcc, html, Input, Output
import plotly.express as px
import seaborn as sns

## Load the Tips dataset
data = sns.load_dataset("Tips")
## Create the dash app
app = dash.Dash(__name__)
app.layout = html.Div([
    html.H1("Tips Dashboard"),
    dcc.Slider(id="bill-slider",
               min=data['total_bill'].min(),
               max=data["total_bill"].max(),
               step=1,
               value=data['total_bill'].mean(),
               marks={i: str(i) for i in range(0,
                                              int(data['total_bill'].max()) + 1, 10)}),
    html.Div([
        dcc.Graph(id='bar-chart'),
        dcc.Graph(id="pie-chart")
    ])
])

```

Implementation Phase: Final Output (no error)



| ASSESSMENT | | | |
|--|-----------|----------------|---------|
| Rubrics | Full Mark | Marks Obtained | Remarks |
| pt | 10 | 10 | |
| ng and Execution/
cal Simulation/ Programming | 10 | 9 | C |
| and Interpretation | 10 | 9 | |
| d of Applied and Action Learning | 10 | 9 | |
| | 10 | 9 | |
| | 50 | 45 | |

of the Faculty

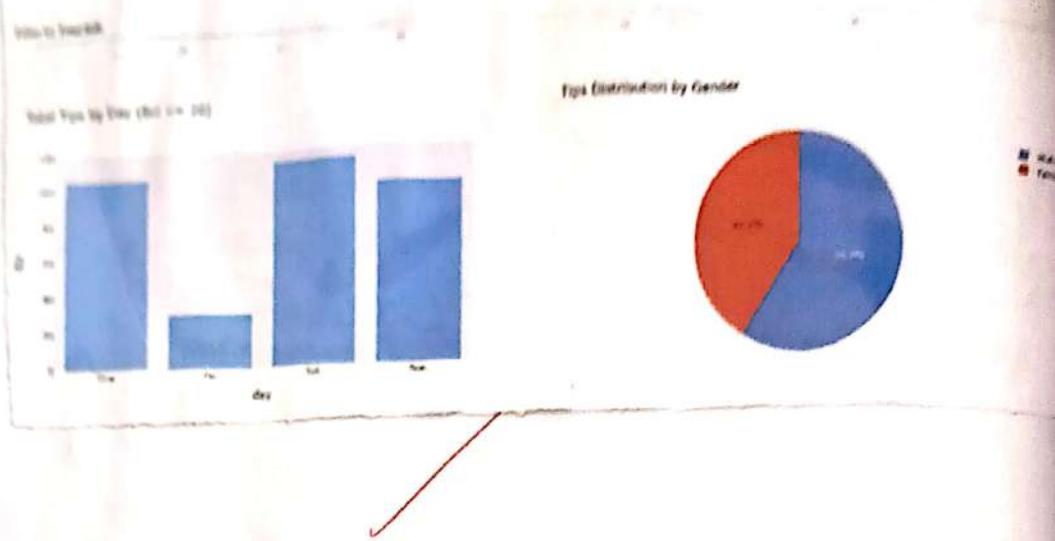
Signature of the Student: M.PremSai

Name : M.PremSai

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

Regn. No. : 241801350008

Customized Tips Dashboard





School: SOBT

Academic Year: 24-25 Subject Name: DATA Campus: VZN

Semester: 1st Program: B.Tech Branch: CSE Subject Code: 1012

Date: 10/10/2024

Applied and Action Learning

(Learning by Doing and Discovery)

of the Experiment: To plot simple & advanced scatter plots

Implementation Phase: Pseudo Code / Flow Chart / Algorithm

1. Simple and advanced Scatter plots using graph and customize them with different shapes and colors.

- (1) Start
- (2) Import libraries
- (3) Load dataset
- (4) Output to jupyter notebook
- (5) Map total bill to colors
- (6) Create bokeh figure
- (7) Add scatter graphs
- (8) Show plot after customizing legend

Page No.....

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



Scanned with OKEN Scanner

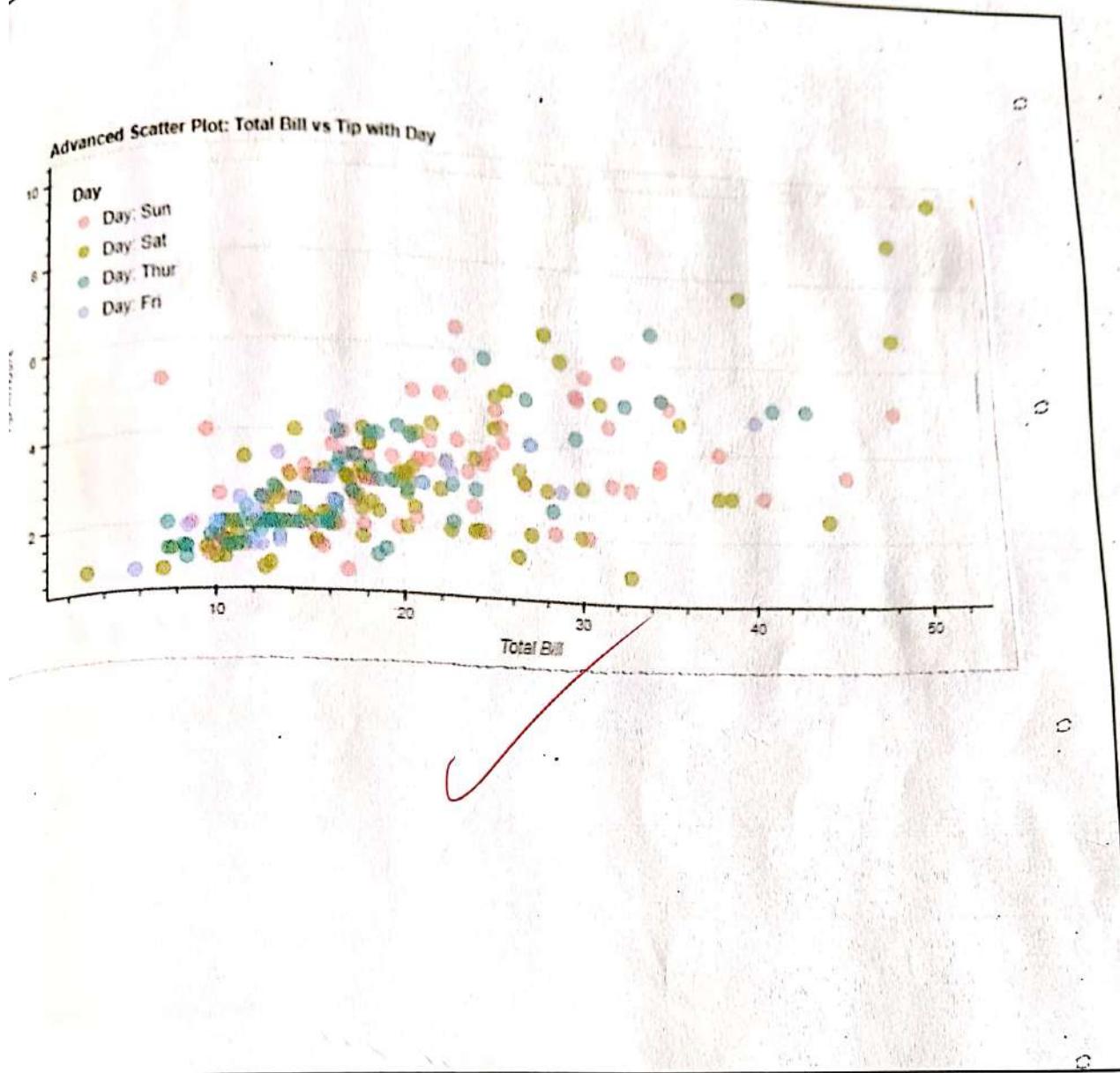
* Testing Phase: Compilation of Code (error detection)

```

# install batch
# pip install batch
from batch import figure show
from batch import output_notebook
import seaborn as sns
# load the tips dataset
tips = sns.load_dataset("tips")
# output to jupyter notebook.
output_notebook()
# map total bill to colors using seaborn and color palette
days = tips['day'].unique()
day_color = {day: color for day, color in
[days, sns.color_palette("husl", len(days))].as_hex}
# create a Bokeh figure
P = figure(title = "Advanced scatter plot",
           x_axis_label = "total Bill vs tip with way",
           y_axis_label = "Tip",
           width = 800, height = 400, tools = "pan",
           wheel_zoom = 700, box = 700, reset)
# Add scatter graphs for each with colors for day in days
subset = tips[tips["day"] == 'day']
P.scatter(x = subset["total_bill"],
          y = subset["tip"],
          color = [day_color(day) for day in range(len(subset))],
          size = 10,
          marker = "circle",
          legend_label = f"way: {day}")
# customize legend
P.legend.location = "top-left"
P.legend.title = "day"
P.legend.title_text_font_style = "bold"
P.legend.background_alpha = 0.6
# show plot
show(P).

```

Implementation Phase: Final Output (no error)



ASSESSMENT

| Rubrics | Full Mark | Marks Obtained | Remarks |
|--|-----------|----------------|---------|
| Implementation and Execution/ | 10 | 10 | |
| Technical Simulation/ Programming | 10 | 9 | |
| Analysis and Interpretation | 10 | 10 | |
| Overall Application of Applied and Action Learning | 10 | 10 | |
| | 10 | 9 | |
| | 50 | 40 | |

Signature of the Student: *M PremSai*

Name : M PremSai *As applicable according to the experimenter
Two sheets per experiment Program Number
Regn. No. : 241601850006

Signature of the Faculty



School: SOFT Campus: VZN
Academic Year: 24-25 Subject Name: DATA Subject Code: 103
Semester: 15 Program: A-Tech Branch: CSE Specialization: CSE
Date:

Applied and Action Learning

(Learning by Doing and Discovery)

of the Experiment:

Doing Phase: Pseudo Code / Flow Chart / Algorithm

- (1) Start
- (2) Import libraries
- (3) Load the tips dataset
- (4) Choose a feature to plot
- (5) Generate HTML content
- (6) Initialize a Dash app
- (7) Run the dash app
- (8) END

Testing Phase: Compilation of Code (error detection)

```

from dash import plotly
from dash import html
from dash import dcc
from dash import Input
from dash import Output
import pandas as pd
import os
# Load the tips dataset
tips = pd.read_csv('tips.csv')
# Choose a feature
feature = "total_bill"
# Create a Bokeh figure
p = Figure(
    title = {"text": "Bokeh Scatter plot for Creature"}, #  

    #tips dataset)
    x_axis_label = "index",  

    y_axis_label = "features",  

    width = 1100,  

    height = 400,  

    background_fill_color = "#ffccf0ff"
)
# Add circles graph
p.circle(x=tips['index'], y=tips[feature],  

    size=10, color="magenta", alpha=0.6,  

    legend_label="Total Bill")
# Generate HTML content for Bokeh plot
html_content = file_html(p, con, "Bokeh plot")
# Initialize Dash app
app = Dash(__name__)
# Layout
app.layout = html.Div([
    html.H1("Dash with Bokeh visualization",
        style = {"text-align": "center"}),
    # Container for bokeh plot
    html.Div(html.Frame(
        src = 'data:text/html,' + html_content,
        style = {"border": "none", "width": "100%",  

            "height": "500px"})),
    # style = {"text-align": "center",  

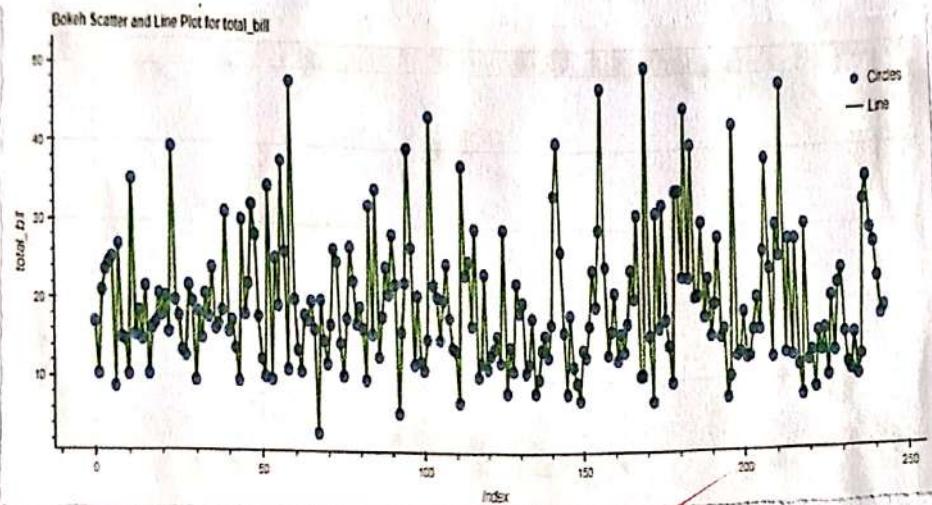
    #         "margin-top": "20px"})
])
# Run the app
if __name__ == "__main__":

```

*As applicable according to
Two sheets per experiment!



Implementation Phase: Final Output (no error)



ASSESSMENT

| Rubrics | Full Mark | Marks Obtained | Remarks |
|---|-----------|----------------|---------|
| pt | 10 | 10 | |
| ing and Execution/
cal Simulation/ Programming | 10 | 9 | |
| and Interpretation | 10 | 10 | |
| I of Applied and Action Learning | 10 | 9 | |
| | 10 | 10 | |
| | 50 | 48 | |

Signature of the Student: M. Prem Saini

Name : M. Prem Saini *As applicable according to the experiment.
Regn. No.: 2019010101 Two sheets per experiment Program No. be used.

of the Faculty

School: soft Academic Year: 2015 Campus: VZM
on ITY Subject Name: DAVP Subject Code: IOP
Semester: 1st Program: B-Tech Branch: CSE Specialization: CN
Date:

Applied and Action Learning

(Learning by Doing and Discovery)

[the Experiment: Test live updating features in dashboard to ensure real time data representation
Phase: Pseudo Code / Flow Chart / Algorithm

Test live updating features in dash board to ensure real-time

- (1) Start
- (2) Import libraries
- (3) Load dataset
- (4) Create visualizations
- (5) Initialize Dash application
- (6) Define layout
- (7) Run the application
- (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)

```

o import dash
from dash import dcc, html
import pandas as pd
import plotly.express as px
import seaborn as sns

o # load the dataset 'tips'
tips_df = sns.load_dataset('tips')
# Create visualization using plotly.
fig1 = px.scatter(tips_df, x='total_bill', y='tip',
color='day', title='total bill vs tip')

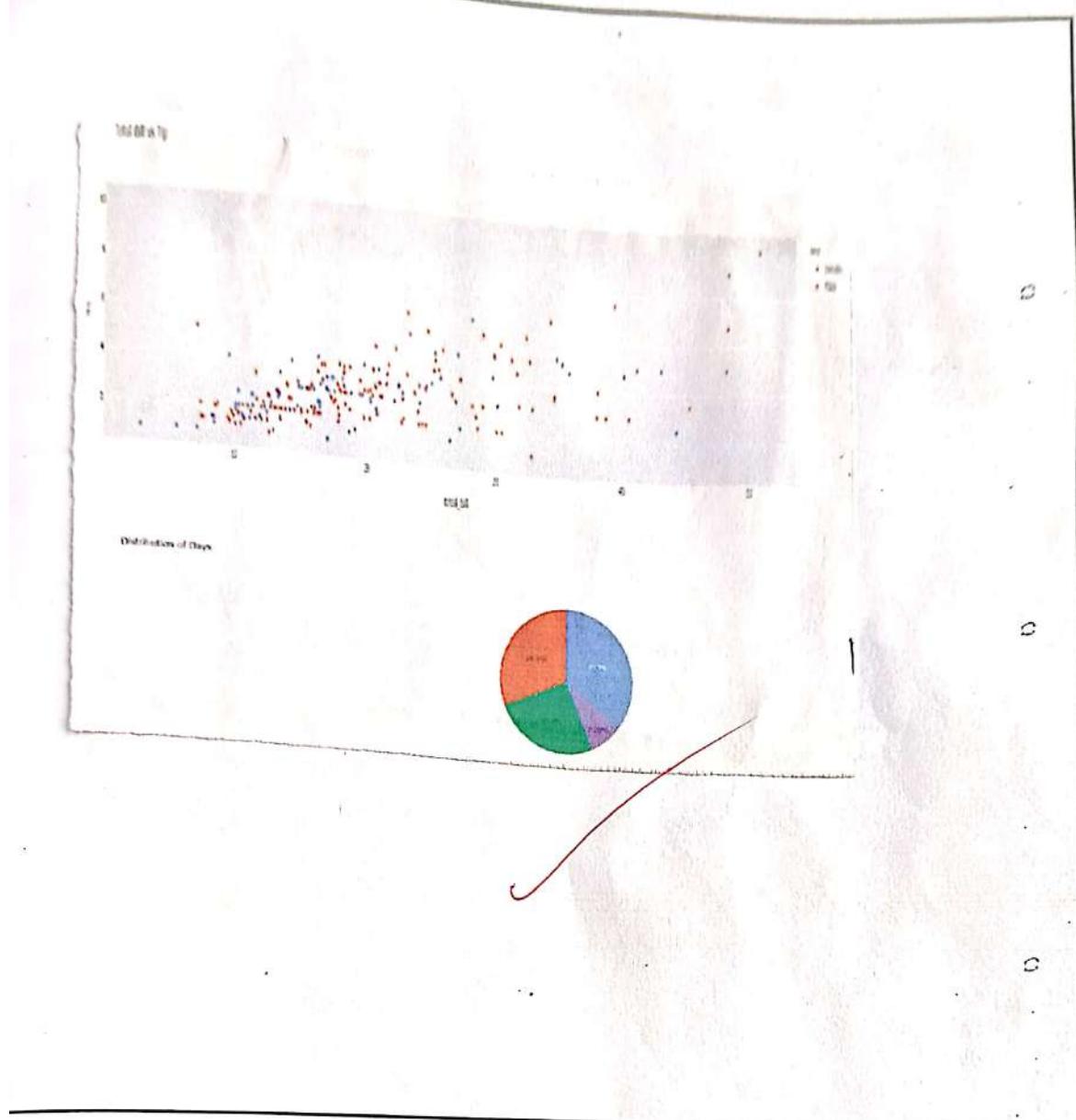
o # pie chart of Day distribution
fig2 = px.pie(tips_df, names='day', title=
"Distribution of tips by Day")

app = dash.Dash(__name__)
app.layout = html.Div([
    html.H1("Interactive tips dataset dashboard"),
    html.Div([
        html.H2("Scatterplot"),
        dcc.Graph(id='scatterplot',
                  figure=fig1)
    ]),
    if __name__ == "__main__":
        app.run_server(port=8050, debug=True)

```

Implementation Phase: Final Output (no error)

Applied and Action Learning



ASSESSMENT

| Rubrics | Full Mark | Marks Obtained | Remarks |
|--|-----------|----------------|---------|
| pt | 10 | 10 | |
| ng and Execution/
cal Simulation/ Programming | 10 | 9 | |
| and Interpretation | 10 | 10 | |
| d of Applied and Action Learning | 10 | 9 | |
| | 10 | 10 | |
| | 50 | 48 | |

of the Faculty

Signature of the Student: M.PremSai

Name : M.PremSai *As applicable according to the experiment.
Regn. No. : 2418036006 Two sheets per experiment. P1020 Note be used.

School: SOBT Academic Year: 19-25 Subject Name: DAVP Campus: VZM
Semester: 1st Program: B-Tech Subject Code: L1118 Branch: CSE Specialization: CSE
Date: 2023-09-11

Applied and Action Learning

(Learning by Doing and Discovery)

of the Experiment :

Implementation Phase: Pseudo Code / Flow Chart / Algorithm

Scrape data from website using web scraping techniques and libraries like BeautifulSoup.

1. Start
2. Import libraries
3. fetch webpage content
4. parse HTML content
5. extract data
6. print URL
7. END.

Page No.....

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



Scanned with OKEN Scanner

* Testing Phase: Compilation of Code (error detection)

Import equality

```
from bs4 import BeautifulSoup  
from requests import get  
  
# Step 1: fetch the webpage content  
URL = "https://www.imdb.com/chart/top"  
# IMDB top movies URL  
  
response = requests.get(url)  
  
# Step 2: parse the HTML content  
soup = BeautifulSoup(response.content, "HTML parser")  
  
# Step 3: extract data  
titles = soup.find_all('td', class_="titleColumn")  
  
for title in titles:  
    print(title.a.text)  
  
ratings = soup.find_all('td', class_="ratingColumn",  
                       mbd_ratings')  
  
for rating in ratings:  
    print(rating.strong.text)  
  
print(URL)
```

Implementation Phase: Final Output (no error)

?

ASSESSMENT

| Rubrics | Full Mark | Marks Obtained | Remarks |
|---|-----------|----------------|---------|
| Experiment | 10 | 10 | |
| Designing and Execution/
Numerical Simulation/ Programming | 10 | 9 | |
| Result and Interpretation | 10 | 10 | |
| Overall Applied and Action Learning | 10 | 9 | |
| | 10 | 9 | |
| | 50 | 42 | |

Signature of the Student: MfremSal

Signature of the Faculty

Name : MfremSal

Regn. No. : 24180135008 *As applicable according to the experiment.
Two sheets per experiment. Page No. 20 Not be used.