



School: S DCT Campus:

Academic Year: 2024 Subject Name: DAVP Subject Code:

Semester: 1 Program: B.TECH Branch: CSE Specialization: CSD

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 corresponding to each category.
- 5) Create bar plot with categories on the x-axis & y-axis.
- 6) set the title of the plot to "bar plot"
- 7) Label the x-axis as "Domain"
- 8) Label the y-axis as "Value"
- 9) Display the plot
- 10) End.

Pseudo code for histplot:-

- 1) Start
- 2) import matplotlib library
- 3) define data list
- 4) Create a histogram with color and edge color
- 5) Set title as "histplot"
- 6) Label Axes
- 7) show plot
- 8) End

Pseudo code for scatterplot:-

- 1) Start.
- 2) import matplotlib library
- 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.



Scanned with OKEN Scanner

* 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-values")
plt.ylabel("y-values")
plt.title("line")
plt.show()

```

compilation of Bar plot

```

import matplotlib.pyplot as plt
categories = ["C1", "C2", "C3", "C4"]
values = [0.5, 1, 1, 0.5, 0.5]
plt.bar(categories, values)
plt.title('Bar Plot')
plt.xlabel('Domain')
plt.ylabel('values')
plt.show()

```

compilation of hist plot:-

```

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

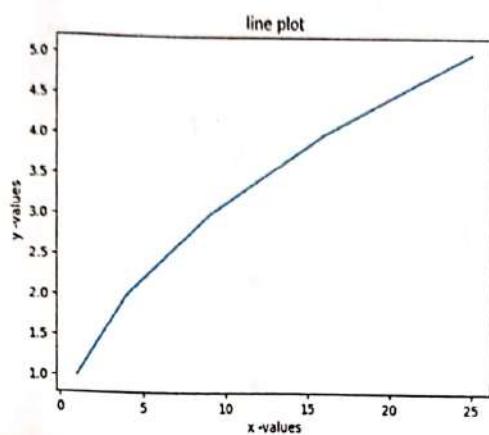
```

compilation of scatter plot

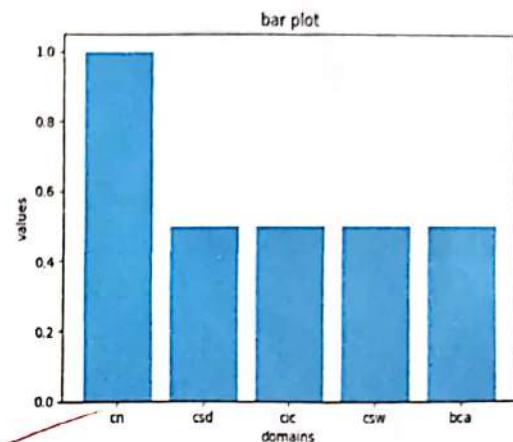
```

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")

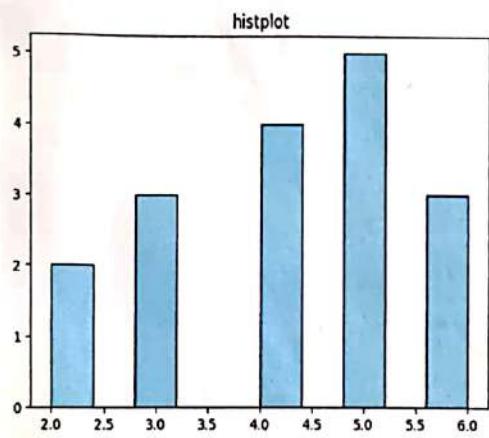
```



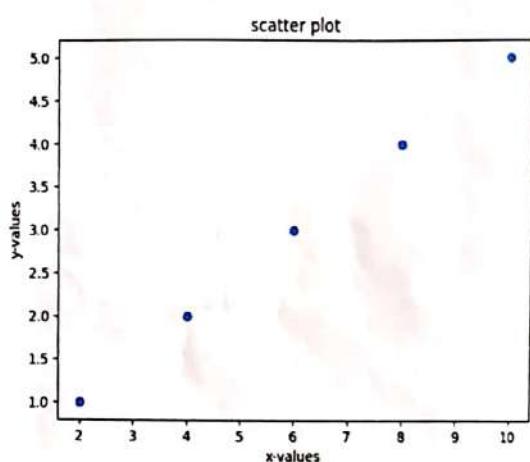
Line representation of data of x, y values



Bar representation of data of domains and its values.



Hist representation of data of the numerical data



Dot representation of data of x, y values.

ASSESSMENT

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

Signature of the Faculty:

Signature of the Student:

Name: S. Koushik

Regn. No.: 241801210022

Page No.....

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



School: S.D.E.T Campus:

Academic Year: 2024 Subject Name: DAVP Subject Code:

Semester: ! Program: B.TECH Branch: CSE Specialization: CSD

Date:

Applied and Action Learning (Learning by Doing and Discovery)

Name of the Experiment:

* Coding Phase: Pseudo Code / Flow Chart / Algorithm

Pseudocode for histogram :-

- 1) Start
- 2) import matplotlib lib
- 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 .

Pseudocode for bar plot :-

- 1) Start.
- 2) import matplotlib lib.
- 3) Define categories and values
- 4) create a bar plot (color & edge color)
- 5) set title as "Barplot"
- 6) set x-label as "Category" and y-label as "Value"
- 7) show plot
- 8) End .

Page No.....

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

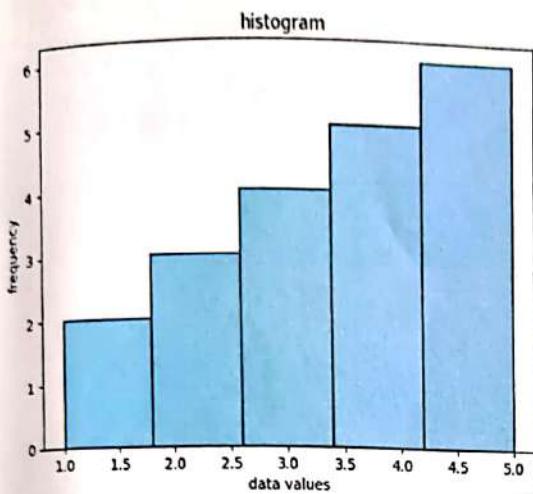


Scanned with OKEN Scanner

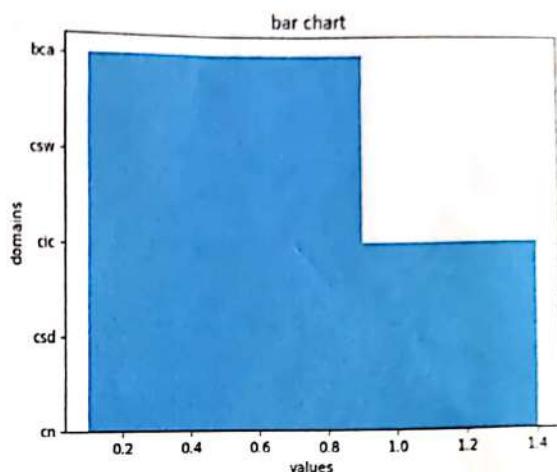
* Testing Phase: Compilation of Code (error detection)

```
import matplotlib.pyplot as plt  
data = [1, 1, 2, 2, 2, 3, 3, 3, 4, 4, 4, 4, 5, 5, 5, 5]  
# create a histogram  
plt.hist(data, bins=5, color="skyblue", edgecolor='black')  
plt.title("Histogram")  
plt.xlabel("Data Values")  
plt.ylabel("Frequency")  
plt.show()
```

```
import matplotlib.pyplot as plt  
# create a data  
categories = ("Aim1", "D", "N", "DCA")  
values = [0.5, 1, 1, 0.5, 0.5]  
# create barplot  
plt.bar(categories, values, color="skyblue", edgecolor="black")  
plt.title("Barplot")  
plt.xlabel("Domain")  
plt.ylabel("Value")  
plt.show()
```



This graph represents the data of numbers



This graph represents the data of domains

ASSESSMENT

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

Signature of the Faculty:

Signature of the Student:

Name: S-Koushik

Regn. No.: 241801080022

Page No.....

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



School: SOCT Campus:

Academic Year: 2024 Subject Name: DAVP Subject Code:

Semester: 1 Program: D.TECH Branch: CSE Specialization: C.S.D.....

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, y₁, y₂ for the Data.
- 3) Create first Area plot b/w x, y₁ with color & transparency.
- 4) Create second area plot b/w x, y₂ by color & transparency.
- 5) Set title for the plot.
- 6) Label x-axis & y-axis.
- 7) Add a legend to differentiable b/w two areas.
- 8) Display the plot.
- 9) Import matplotlib library.
- 10) Create a list for labels, size, color and
- 11) Create pie chart with (size, explode, labels, percentage, shadow).
- 12) Set the title for the plot
- 13) 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)

~~# Create Area plot.~~

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

```
x = [1, 3, 5, 7, 9]
```

```
y1 = [2, 4, 1, 11, 10]
```

```
y2 = [1, 3, 5, 7, 9]
```

```
plt.fill_between(x, y1, color="indigo", alpha=0.4,  
label="y1", edgecolor="black")
```

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

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

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

```
plt.legend()
```

Labels ("chocolate", "vanilla", "strawberry",
"pistachio")

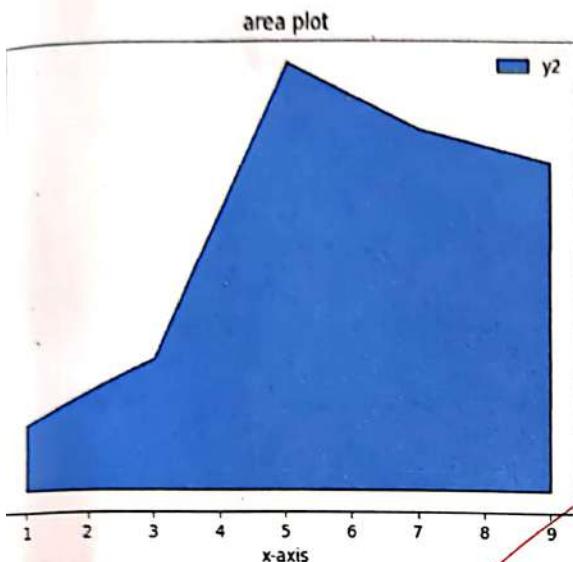
~~Sizes = [50, 25, 20, 15]~~~~Colors = ["brown", "white", "pink", "green"]~~~~Explode = [0.1, 0, 0, 0]~~

```
# Create pie chart (sizes explode = explode, labels = labels)
```

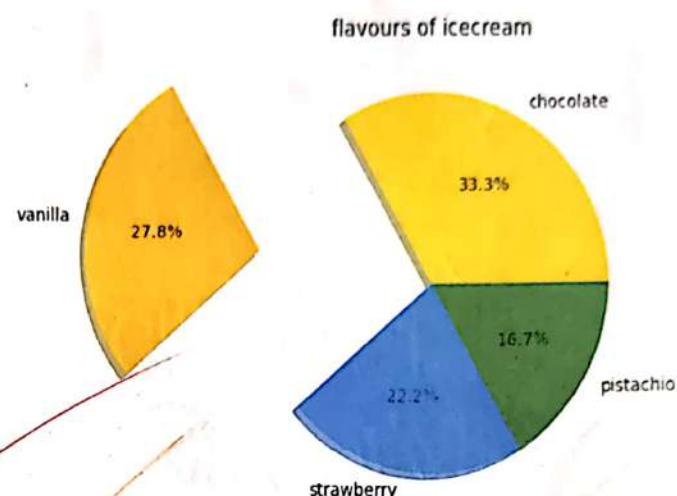
```
plt.title("flavours of ice cream")
```

```
plt.show()
```

Shadow=(T)



The graph represents area plot data



The graph represents the flavours of ice cream data

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	06	
Total	50	41	

Signature of the Student:

Name: S. Koushik

Regn. No.: 241301970022

Page No.....

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

Signature of the Faculty:



School: SOGT Campus:

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

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

Date:

Applied and Action Learning (Learning by Doing and Discovery)

Name of the Experiment:

Coding Phase: Pseudo Code / Flow Chart / Algorithm

Pseudocode for Pie plot:-

- 1) Start
- 2) Import matplotlib
- 3) Define categories and values
- 4) Define Explode and colors
- 5) Create pie chart with (shadow, explode, colors)
- 6) Set title as "Pie plot of domains"
- 7) Show plot
- 8) End.

Pseudocode for Bar plot :-

- 1) Start
- 2) Import matplotlib
- 3) Define categories list
- 4) Define values list
- 5) Create bar plot with (color, edge color)
- 6) Set title as "Bar plot of Domains"
- 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.



Scanned with OKEN Scanner

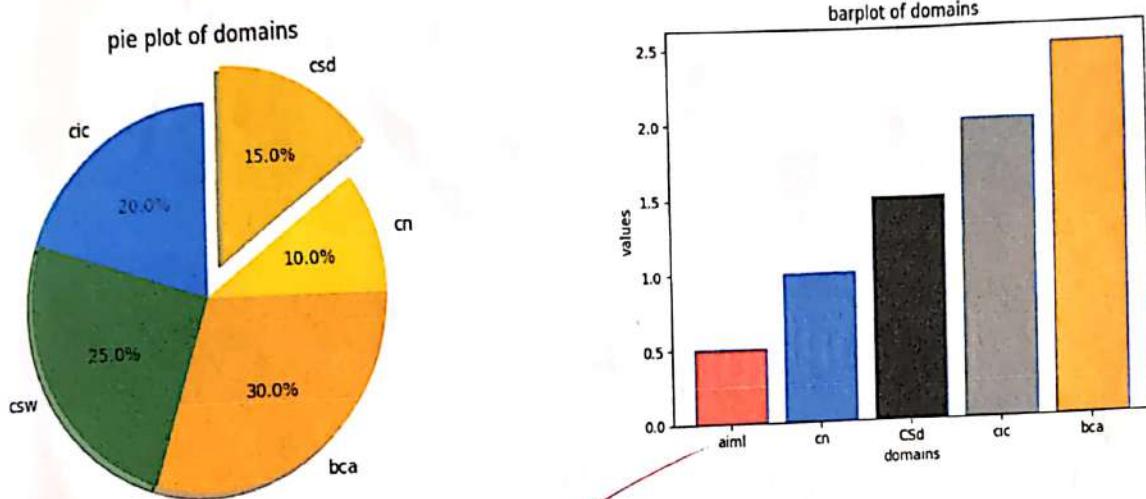
Testing Phase: Compilation of Code (error detection)

```

import matplotlib.pyplot as plt
categories = ["Aiml", "Os", "St", "Cn", "Bca"]
values = [10, 20, 30, 40, 50]
explode = [0, 0.2, 0, 0, 0]
colors = ["yellow", "red", "yellow green", "blue",
          "light coral", "light sky blue"]
plt.pie(values, labels=categories, autopct='%1.1f%%',
         shadow=True, explode=explode,
         title="Pie plot of domains")
plt.show()
    
```

```

import matplotlib.pyplot as plt
categories = ("Aiml", "Os", "St", "Cn", "Bca")
values = [0.5, 1, 1.5, 2, 2.5]
color = ("red", "blue", "black", "gray", "orange")
plt.bar(categories, values, color=color, edgecolor="black",
         title="Bar plot of Domains")
plt.xlabel("Domains")
plt.ylabel("Values")
plt.show()
    
```



This graph representing the percentage of students present in each domain

This graph representing 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	10	
Record of Applied and Action Learning	10	10	
Viva	10	10	
Total	50	49	

Signature of the Faculty:

Signature of the Student:

Name: S. Koushik

Regn. No.: 241801080622 Page No.....

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



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 matplotlib
- 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.

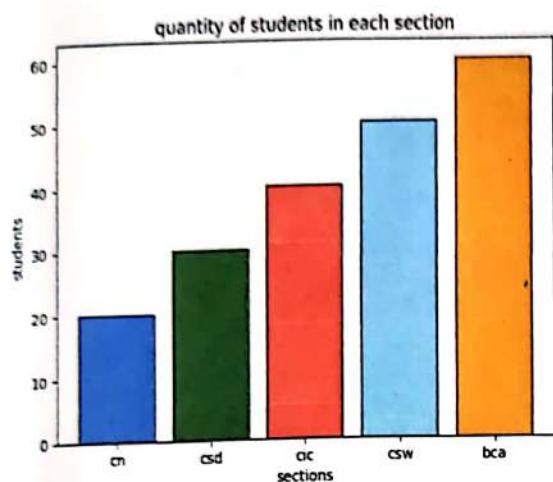
Pseudo code for scatter plot:-

- 1) import matplotlib
- 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

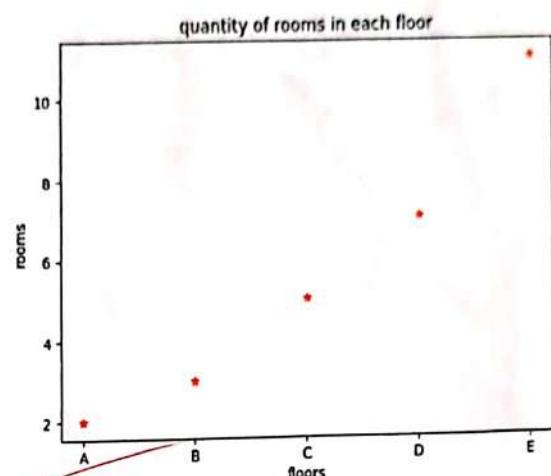
* Testing Phase: Compilation of Code (error detection)

```
import matplotlib.pyplot as plt  
categories = ["A", "B", "C", "D", "E"]  
values = [10, 20, 30, 40, 50]  
color = ["blue", "green", "red", "skyblue", "purple"]  
plt.bar(categories, values, color=color, edgecolor="black")  
plt.xlabel("sections")  
plt.ylabel("Students")  
plt.title("Quantity of students 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()
```



This graph represents no of students present in each section



This graph represents no. of rooms present in each floor

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	09	
Viva	10	10	
Total	50	47	

Signature of the Student:

Name: S. Koushil

Page No.....

Regn. No.: 24180118002

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

Signature of the Faculty:



School: SOCI Campus: V-2 M

Academic Year: 2024-25 Subject Name: D.A.V.P. Subject Code: 1018

Semester: IST 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

Pseudo Code - To read CSV file:-

- 1) Import Pandas as pd
- 2) Read the CSV file into a data frame using pd.read_csv
- 3) Print the data frame

* Testing Phase: Compilation of Code (error detection)

```
# Read Csv file
```

```
import Pandas pd
```

```
df = pd.read_csv("Kouclick.csv")
```

```
Print(df)
```

S. NO	Names	Marks
1	Kohli	100
2	Dhoni	70
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	09	
Result and Interpretation	10	09	
Record of Applied and Action Learning	10	10	
Viva	10	10	
Total	50	48	

Signature of the Faculty: 

Signature of the Student: S. Koushik

Name: S. Koushik

Regn. No.: 241801380022 Page No.....

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



School: SDET Campus:

Academic Year: 2024 Subject Name: P.A.V.P. Subject Code:

Semester: Program: R-TECH. Branch: CSE Specialization: CSD.....

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 lib pyplot, & numpy libraries.
- 2) create a data frame with "data" column "name" "marks" & "sinθ"
- 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 lib pyplot and numpy libraries.
- 2) Create a boxplot with "name" on x-axis and mark 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("Roushik.csv")
```

```
print(data).
```

```
* visualise 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  
data = data, alpha=1)
```

```
plt.show()
```

```
# To visualize data plot.
```

```
import seaborn as sns
```

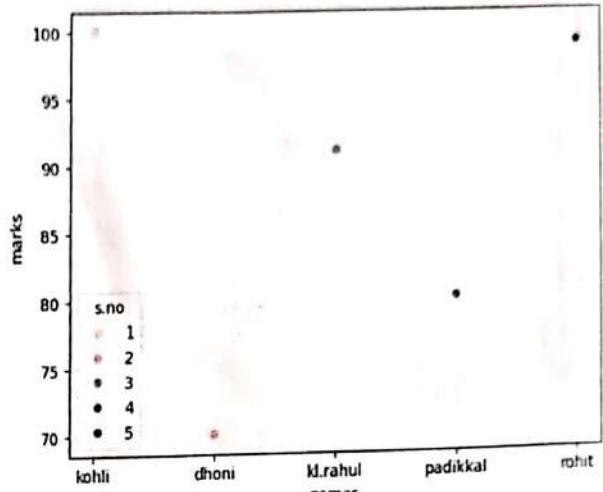
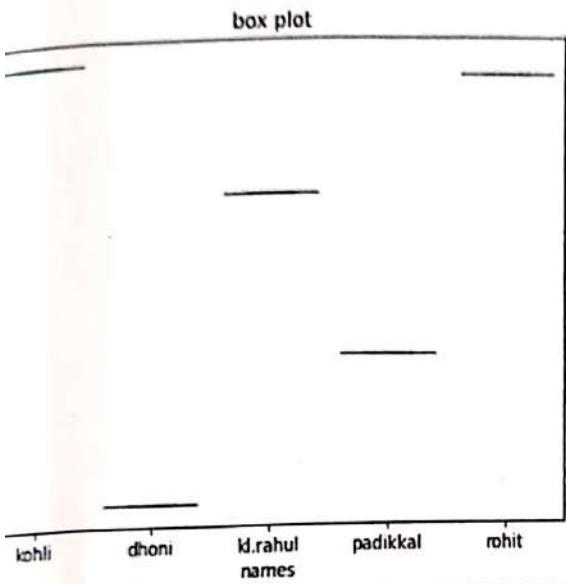
```
import matplotlib.pyplot as plt
```

```
import numpy as np.
```

```
sns.boxplot(x="Name", y="Marks", data
```

```
plt.title("Boxplot")
```

```
plt.show()
```



This CSV file represents cricketers in a box plot.

↳ This CSV file represents cricketers in a scatter plot.

ASSESSMENT

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

Signature of the Student:

Name: S. KOUZHIL

Regn. No.: 241301380022 Page No.....

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

Signature of the Faculty:



School: S.P.G.T Campus:

Academic Year: 2024 Subject Name: D.A.V/I Subject Code:

Semester: Program: B.TECH Branch: CSE Specialization: Z.S.D.....

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 & 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.

Pseudocode for histogram:-

- 1) import matplotlib.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 and y-axis.
- 6) Display the plot.

Pseudo code for Bar chart:-

- 1) import matplotlib.pyplot as plt
- 2) Create lists for categories & vals.
- 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()
```

```
# Line plot.
```

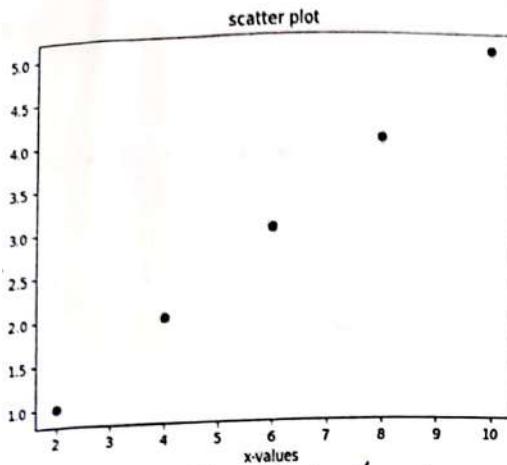
```
import matplotlib.pyplot as plt  
x = [1, 2, 3, 4, 5]  
y = [1, 4, 9, 16, 25]  
plt.plot(x, y, marker="o", color="black", linewidth=2)  
plt.xlabel("x-values")  
plt.ylabel("y-values")  
plt.title("Line chart")  
plt.show()
```

```
# histogram
```

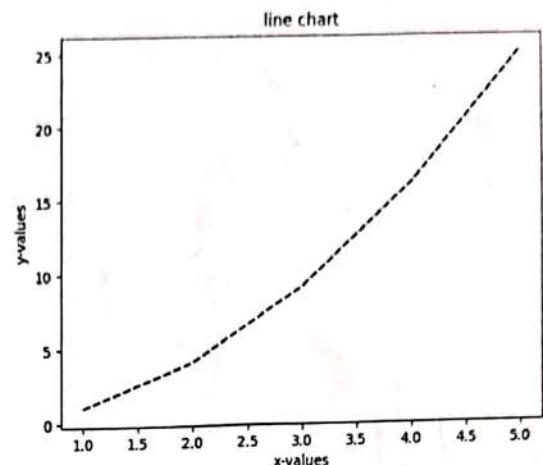
```
data = [1, 1, 2, 2, 2, 3, 3, 3, 3, 4, 4, 4,  
        4, 4, 5, 5, 5, 5, 5, 5, 5]  
plt.hist(data, color="sky blue", edgecolor="black")  
plt.title("Histogram")  
plt.xlabel("Values")
```

```
# Bar chart.
```

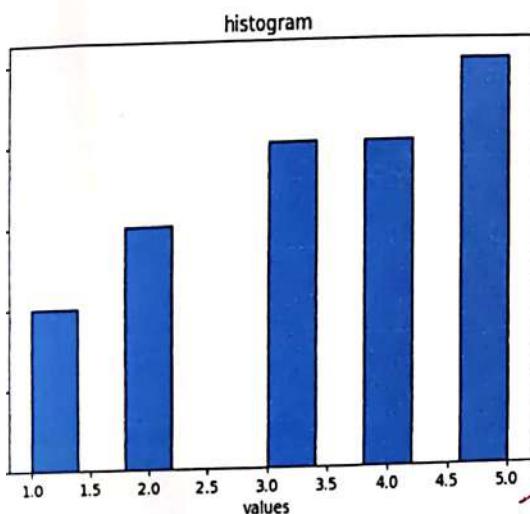
```
categories = ["Aim1", "Cic",  
              "CSB"]  
values = [1, 2, 3, 4, 5]  
plt.bar(values, categories)  
plt.title("Bar chart")  
plt.xlabel("Values")  
plt.ylabel("Domain")
```



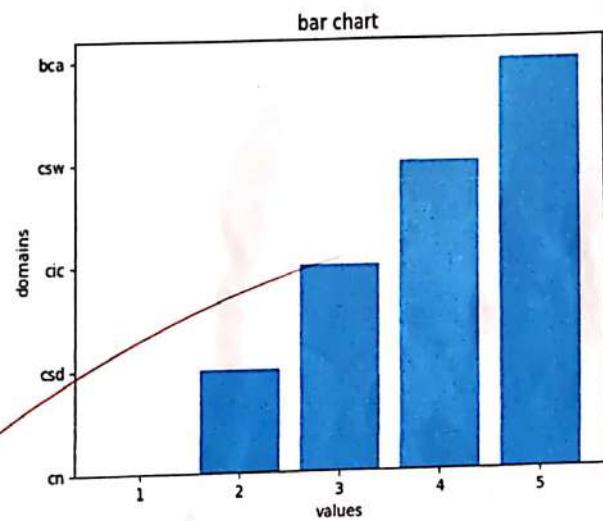
Output of scatter plot :- of x, y values



Output of line chart :- of various numbers



Output of histogram of numerous values.



Output of Bar chart of domains and its values.

ASSESSMENT

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

Signature of the Student:

C. MUNIRUL



Scanned with OKEN Scanner



Centurion
UNIVERSITY
Engineering
Commissions

School: SOET Campus:

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

Semester: 1 Program: 1B-TGCP Branch: CSE Specialization: CSD

Date:

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

Coding Phase: Pseudo Code / Flow Chart / Algorithm

Pseudo code for bubble chart:-

- 1) Import matplotlib.pyplot
- 2) Create lists x, y and size for coordinates of 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.

Pseudo code for heatmap

- 1) Import the seaborn, matplotlib.pyplot & numpy lib.
- 2) Create 10x10 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.....

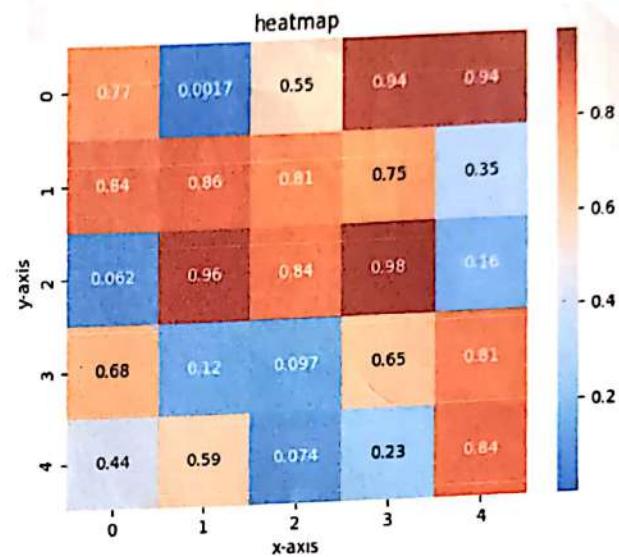
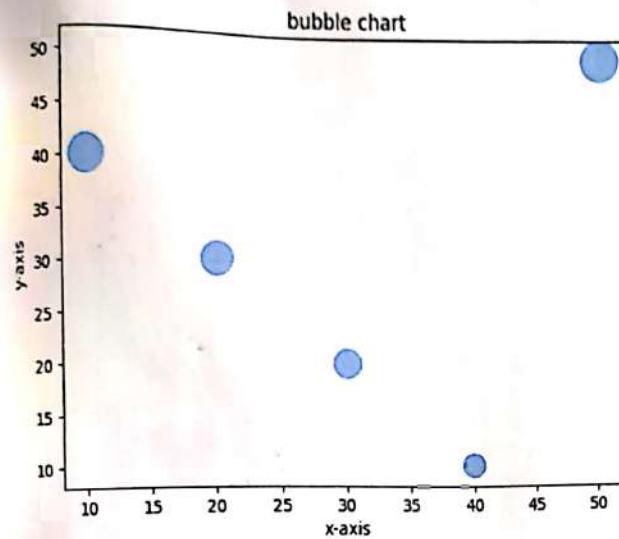
*As applicable according to the experiment.
Two sheets per experiment (10-20) to be used.



Scanned with OKEN Scanner

* 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("Heatmap")  
plt.xlabel("x-axis")  
plt.ylabel("y-axis")  
plt.show()
```



The graph represents the bubble chart
data : of x, y values.

The graph represents data about heat map

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	10	
Total	50	48	

Signature of the Student:

Name: S.KUSHLIK

Page No.....





Centurion
UNIVERSITY
Sri Lanka

School: Campus:

Academic Year: Subject Name: Subject Code:

Semester: Program: Branch: Specialization:

Date:

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment :

* Coding Phase: Pseudo Code / Flow Chart / Algorithm

- ① start.
- ② pip 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.



Scanned with OKEN Scanner

* Testing Phase: Compilation of Code (error detection)

① Text styles.

```
style = {"text-align": "center", "color": "#4CAF50", "font-size": "16px", "font-family": "Arial", "margin-bottom": "10px"};
```

② Dropdown style.

```
style = {"width": "150px", "display": "inline-block", "border": "1px solid #4CAF50", "border-radius": "5px", "margin-right": "5px"};
```

③ Styling.

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

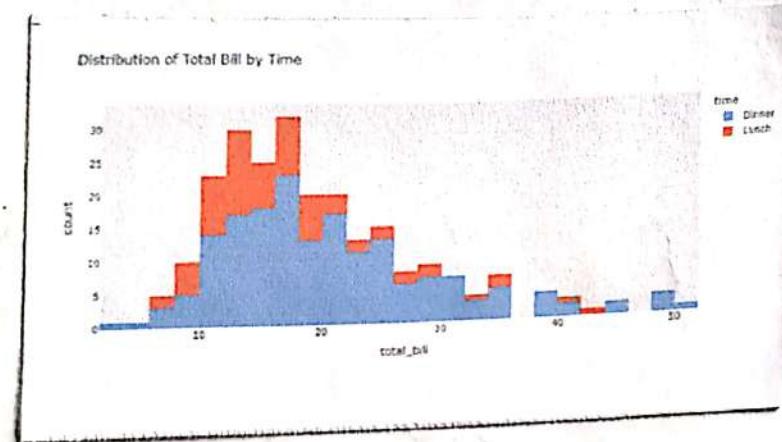
④ Background.

"background-color": "#ffffdd"

⑤ Box shadowing.

"box-shadow": "0 4px 8px rgba(0,0,0,0.2)"

* Implementation Phase: Final Output (no error)



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

Signature of the Student:

Name :

Regn. No. :

*As applicable according to the experiment.
Two sheets per experiment. Please Note.

Signature of the Faculty



School: Campus:
Academic Year: Subject Name: Subject Code:
Semester: Program: Branch: Specialization:
Date:

Applied and Action Learning

(Learning by Doing and Discovery)

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)

Search functionality.

```
dcc.dropdown (
```

```
    id = 'x-axis', # unique identifier for callback
```

```
    options = [{label: col, value: col} for col in
```

```
tips - df.columns],
```

```
    value = 'total_bill' # Default
```

```
    style = {"width": "100%", "display": "block",  
             "margin-right": "5px"}  
)
```

```
html.Div (
```

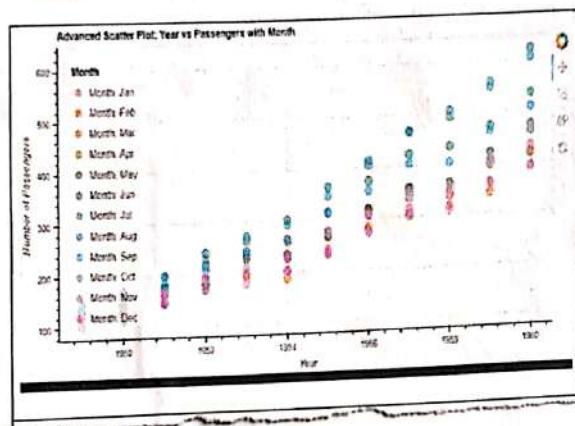
```
    html.Label ("Select year (if applicable):",
```

```
    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	10	10	
Total	50	47	

Signature of the Student:

Name :

Regn. No. :

*As applicable according to the experiment.
Two sheets per experiment. Program No. to be used.

Signature of the Faculty



School: Campus:

Academic Year: Subject Name: Subject Code:

Semester: Program: Branch: Specialization:

Date:

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

* Coding 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 (---)
- 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 data set.
df = sns.load_dataset("tips")
app = Dash(__name__)

# layout.
app.layout = html.Div([
    html.H1("Tips Dataset Dashboard"),
    # style="text-align:center"
])

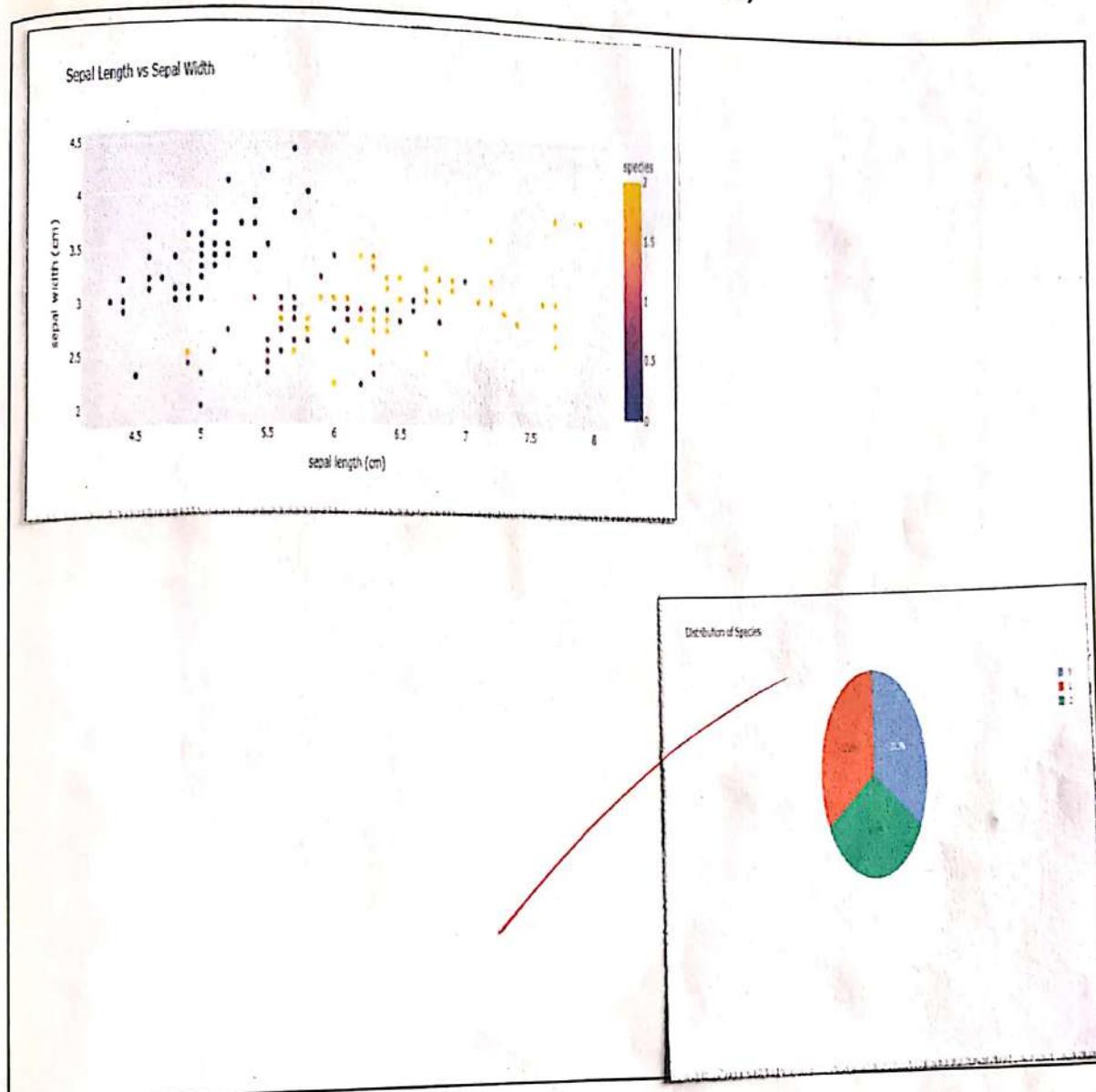
# Dropdown to select column.
html.Label("Select x-axis for visualization"),
dcc.dropdown(
    id='x-axis',
    options=[{'label': col, 'value': col}]

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

# Bar chart
output('bar chart', figure)
input('x-axis', 'value'),
    def update_pie_column():
        return px.pie(title="invalid student",
                      # run in app
                      if __name__ == '__main__':
                        app.run_server(port=8055, debug=True).

```

* 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	9	
Viva	10	10	
Total	50	46	

Signature of the Faculty

Signature of the Student:

Name :

Regn. No. :

*As applicable according to the experiment.
Two sheets per experiment. Page No. _____ Note used.



School: Campus:

Academic Year: Subject Name: Subject Code:

Semester: Program: Branch: Specialization:

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 file_html.
- (4) from bokeh import con.
- (5) from dash import Dash
- (6) from dash import html, dcc.
- (7) from dash import input, output
- (8) from import pandas as pd
- (9) import seaborn as sn
- (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.H1("Dash with Bokeh visualization")

style = {"text-align": "center"},

value = 'total bill'

),

], style={"width": "50%", "margin": "auto"})

@app.callback(...)

output ['bokeh plot', 'children'])

input ['bokeh feature', 'value'])

P = figure (

title = "Bokeh scatter and plot for (feat)"

x-axis = label = 'Index'

input ['bokeh - feat'] . value

P. circle (x = tips['df.index'], y = tips['df[feat]'] ,

size = 10, color = "gray", alpha = 0.6, legend = label)

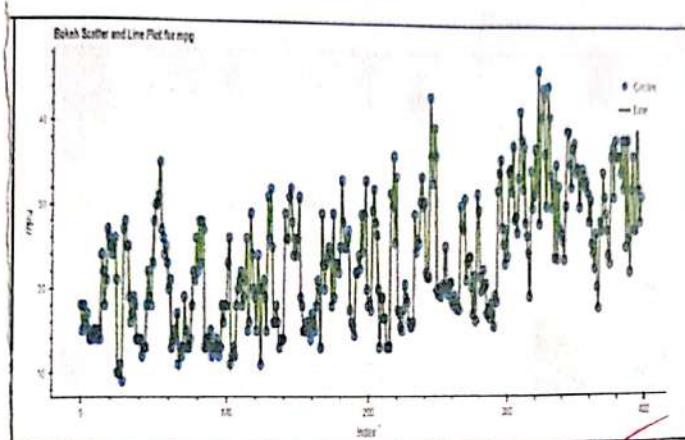
html content = file.html (P, con, "Bokeh plot")

run app.

if __name__ == '__main__':

app.run_server (port = debug = true)

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. Page No. _____.

Signature of the Faculty



School:

Academic Year: Subject Name: Campus:

Semester: Program: Subject Code:

Date: Branch: Specialization:

Applied and Action Learning

(Learning by Doing and Discovery)

Name of the Experiment:

* Coding Phase: Pseudo Code / Flow Chart / Algorithm

- ① start.
- ② Import 'plotting.figure' , show from bokeh.
plotting
- ③ import output_notebook from bokeh .
- ④ import column Datasource from bokeh . models .
- ⑤ import seaborn as sns.
- 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



Scanned with OKEN Scanner

* Testing Phase: Compilation of Code (error detection)

After loading dataset tips

```
sex_color = ("Male": "blue", "Female": "red")
```

```
smoke_shape = ("Yes": "triangle", "No": "circle")
```

P = figure (title = "Advanced scatter plot", total_bill
with sex and smoker", x-axis-label = "total_bill",
y-axis-label = "tip(%)")

for smokers, shape in smoke_shape.items():

P.scatter(

```
subset {"total_bill"}]
```

```
subset ("tip")
```

```
marker = shape
```

```
size = 15,
```

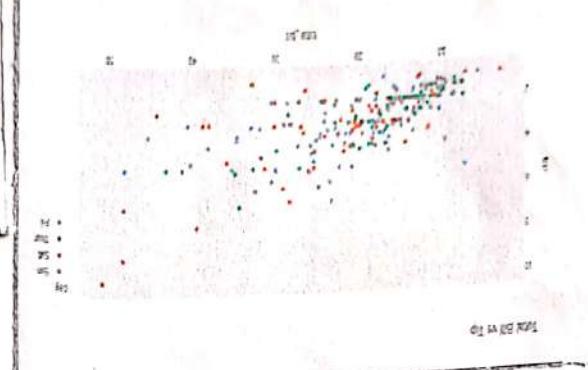
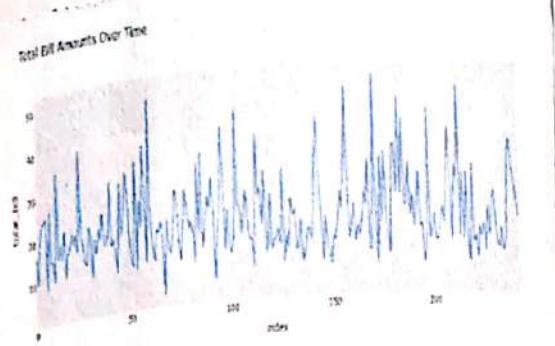
```
legend-label = (smoker, non-smoker)
```

P.legend + location = "top-left"

P.legend title = "legend"

show (P).

* Implementation Phase: Final Output (no error)



Rubrics	ASSESSMENT		Remarks
	Full Mark	Marks Obtained	
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. Page No. _____ be used.

Signature of the Faculty



School: Campus:
Academic Year: Subject Name: Subject Code:
Semester: Program: Branch: Specialization:
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 \$ beautiful soup.
- 2) # Fetch web pages content.-
→ Define URL & send GET request.
- 3) # Parse HTML content.
→ Parse content with beautiful soup.
- 4) Extract Data.
 - Find & print movie titles.
 - Find & print movie ratings.
- 5) Print URLs.
→ Print the URLs.

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 requests  
from bs4 import BeautifulSoup  
  
# Fetch the webpage content  
URL = "https://www.imdb.com/chart/top"  
response = requests.get(URL)  
  
# Parse the HTML content  
soup = BeautifulSoup(response.content, 'html.parser')  
  
# Extract data  
titles = soup.find_all('div', class_='titleColumn')  
for title in titles:  
    print(title.a.text)  
  
# Extract all the movie ratings  
ratings = soup.find_all('div', class_='ratingColumn rating')  
print(ratings[0].strong['text'])  
print(ratings[1].strong['text'])  
print(ratings[2].strong['text'])
```