



Government Engineering College

Sec-28 Gandhinagar

Sem: - VI

Subject: - Data Visualization

Subject Code: - 3160717



Government Engineering College

Sec-28 Gandhinagar

Certificate

This is to certify that

Mr./Ms. Of class

..... Division, Enrollment No. Has

Satisfactorily completed his/her term work in

..... Subject for the term ending in

..... 2021.

Date: -

Contents

1. Institute Vision/Mission.....	4
Vision:	4
Mission:.....	4
2. Computer Engineering Department Vision/Mission	5
Vision:	5
Mission:.....	5
3. Program Educational Outcome (PEO)	6
4. PSO.....	6
5. POs	6
6. Submission Instructions:.....	8
7. Assignment Index.....	9
8. Practical Index	9
9. Assignment 1	10
10. Assignment 2	Error! Bookmark not defined.
11. Assignment 3	Error! Bookmark not defined.
12. Assignment 4	Error! Bookmark not defined.
13. Practical 1	Error! Bookmark not defined.
14. Practical 2	Error! Bookmark not defined.
15. Practical 3	Error! Bookmark not defined.
16. Practical 4	Error! Bookmark not defined.
17. Practical 5	Error! Bookmark not defined.

Institute Vision/Mission

Vision:

- To be a premier engineering institution, imparting quality education for innovative solutions relevant to society and environment.

Mission:

- To develop human potential to its fullest extent so that intellectual and innovative engineers can emerge in a wide range of professions.
- To advance knowledge and educate students in engineering and other areas of scholarship that will best serve the nation and the world in future.
- To produce quality engineers, entrepreneurs and leaders to meet the present and future needs of society as well as environment.

Computer Engineering Department

Vision/Mission

Vision:

Mission:

Program Educational Outcome (PEO)

- To provide students with a strong foundation in the mathematical, scientific and engineering fundamentals necessary to formulate, solve and analyze engineering problems and to prepare them for graduate studies, R&D, consultancy and higher learning.
- To develop an ability to analyze the requirements of the software, understand the technical specifications, design and provide novel engineering solutions and efficient product designs.
- To provide exposure to emerging cutting edge technologies, adequate training & opportunities to work as teams on multidisciplinary projects with effective communication skills and leadership qualities.
- To prepare the students for a successful career and work with values & social concern bridging the digital divide and meeting the requirements of Indian and multinational companies.
- To promote student awareness on the life-long learning and to introduce them to professional ethics and codes of professional practice

PSO

By the completion of Computer Engineering program the student will have following Program specific outcomes.

- Design, develop, test and evaluate computer-based systems by applying standard software engineering practices and strategies in the area of algorithms, web design, data structure, and computer network
- Apply knowledge of ethical principles required to work in a team as well as to lead a team

POs

Engineering Graduates will be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
2. **Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of

mathematics, natural sciences, and engineering sciences.

3. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
4. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
5. **Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modelling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Data Visualization (3160717)

Course Outcomes (COs)

CO-1	
CO-2	
CO-3	
CO-4	

Submission Instructions:

1. Assignments' submission is **Handwritten**
2. Practical submission is **Computerized**
 - a. **Add** Practical No., Aim, Solution Algorithm and Description
 - b. **Add** Executable Code with necessary comments
 - c. **Add** Output snapshots having Practical No., Aim, Student Enrolment No, Date & time of execution, and practical input-output.
3. Take the print out (double sided) of this document with practical details and attach your handwritten assignment in it.
4. Update the index.

7. Assignment Index

Sr. No	Assignment	Date	Page No.	Sign
1	Assignment 1	25/11/2022		
2	Assignment 2	20/01/2022		
3	Assignment 3	24/02/2022		
4	Assignment 4	30/3/2022		

8. Practical Index

Sr. No	Assignment	Date	Page No.	Sign
1	Practical 1	24/11/2022		
2	Practical 2	19/01/2022		
3	Practical 3	23/02/2022		
4	Practical 4	16/03/2022		
5	Practical 5	06/04/2022		

9. Assignment 1

CO1: Explain principles of visual perception.

Module 1

1. What is data visualization.
2. List the charts that we can use to visualize different data.
3. Explain how can we acquire & represent data simultaneously?
4. Explain any three charts to visualize the data in details
5. Write Different applications of Data Visualization.
6. Explain key factors of Data visualization.
7. Explain HTML5 key features.
8. What is infographic? Explain with example.
9. Explain the key principles of infographic.
10. What is HTML5 canvas? Explain with example.
11. What is SVG? Explain with example.

* Assignment - 1

1) What is data visualization?

- Data visualization is graphical representation of information and data by using visual elements like charts, graphs and maps. Data visualization tools provide an accessible way to see and understand trends, outliers and patterns in data.

2) List the charts that we can use to visual different data.

-
- Column chart
 - Bar chart
 - Stacked Bar chart
 - Stacked column chart
 - Area chart
 - Dual Axis chart
 - Line graph
 - Pie chart
 - Scatter plot chart

3) Explain how we can acquire and represent data simultaneously?

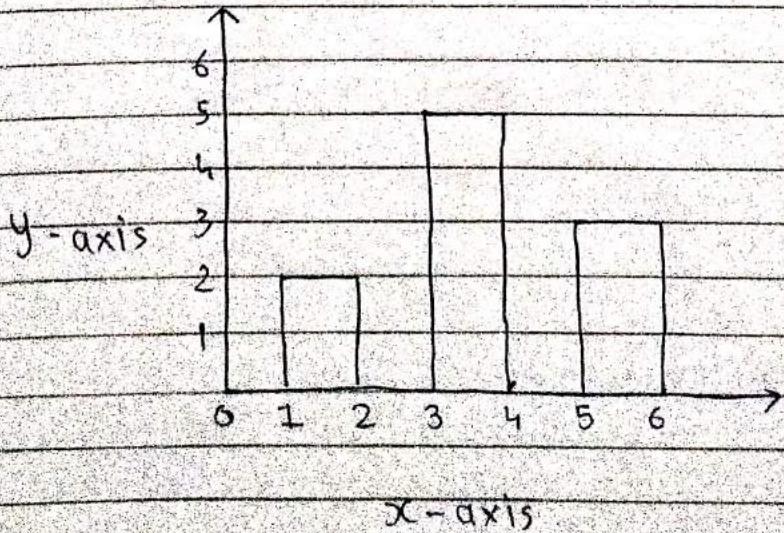
-
- The world of data visualization doesn't just consume existing data new data is constantly being added to the store in real time
 - Information can be collected directly through an HTML form on a website and incorporated into representation

- Collecting live data has many challenges, but the recent advances made by the widespread acceptance of HTML5 have ameliorated many of them
- When combined with a few javascript libraries, it's now possible to use advanced term elements

4) Explain any three charts to visualize data.

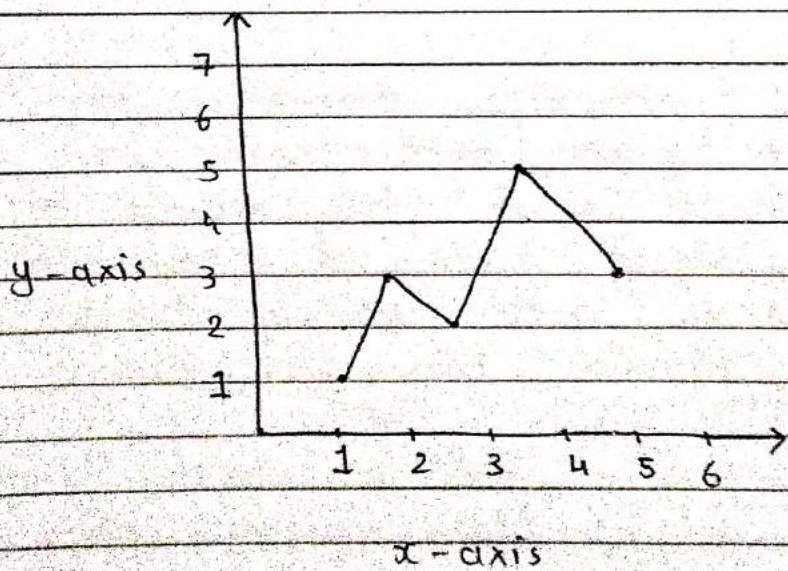
→ * Column chart:-

- Column chart use vertical columns to show numerical comparisons between categories and the number of column's should not be large.
- The column chart takes advantage of the height of the column to restrict the difference in the data, and the human eye is sensitive to height difference. The limitation is that it is only suitable for small and medium sized data sets.
- Application:- Comparison of classified data.



* Line chart:-

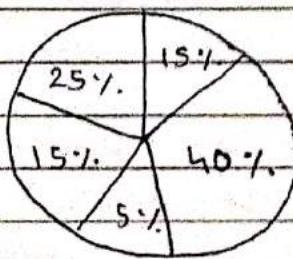
- A line chart is used to show the change of data over a continuous time interval or time span. It is characterized by a tendency to reflect things as they change over time or ordered categories.
- It should be noted that number of data records of the line graph should be greater than 2, which can be used for trend comparison of large data volumes and it is better not to exceed 5 polylines of the same graph.
- Application:- Trend of data volume over time, series trend



* Pie chart:-

- Pie chart are widely used in various fields to represent the proportion of different classification, and to compare various classification by the arc.

- The Pie chart is not suitable for multiple series of data, because as the series increase, each slice becomes smaller, and finally the size distinction is not obvious.
- A Pie chart can also be made into a multi-layer pie-chart, showing the proportion of different categorical data, while also reflecting the hierarchical relationship.
- Application:- Series, ratio, series size comparison.



- 5) Write different application of data visualization
-
- Data visualization application enables users to visualize data, draw insights and understand it better. It allows people to organize and present pictures better than tables that contain rows and columns.
 - By visualizing data, you are able to ask more questions and quickly find answers.
 - Data visualization tools provide valuable decision support by allowing users to quickly create and modify dashboards.

6) Explain key features of data visualization.

- - 1. Determine the best visual.
 - 2. Balance the design
 - 3. Focus on the key areas
 - 4. Keep it simple
 - 5. Corporate interactivity
 - 6. User patterns
 - 7. Compare aspects
 - 8. Summary

7) Explain HTML5 key features.

- - These are a lot of things and tags available in HTML.

* doctype:- you need not to write down a long and messg doctype tag just type
`<!DOCTYPE html>`

* structured tags:-

You now have `<section>`, `<header>`, `<main>`, `<footer>`, `<nav>`, `<menu>`, `<article>`, `<figure>` to structure your markup well.

* Form fields:-

You have more options than just the text or password. listing date, time, week, time, number you can do more with the forms with HTML5

* Media elements:-

You now have `<audio>` and `<video>` tags to embed rich media without having to write a messg code. Both make use of `<source>` tag within.

* Canvas :-

- <canvas> tag allows you to add a canvas - like space to webpages. handy to pull dynamic graphics with JS you can bring it one step further by adding a JS - driven game

8) What is infographic? Explain with example.

- Infographic is combination of two words information and graphic.
- Infographic is a visual representation of information of data.
- An infographic is a collection of imagery charts and minimal text that gives an easy to understand overview of a topic.
- Infographic use striking, engaging visuals to communicate information quickly and clearly.
- It provides a quick overview of a topic, explain a complete process, display research finding or survey data.

9) Explain the key principle of infographic.

- To create stunning, easy to understand, good infographic we can follow the basic principles.
- Emphasize:- When you pop-out the most important element, it means emphasize.

- Hierarchy:- Infographic design should have a hierarchy to control the user flow when reading our infographic. It should be most important to less important.

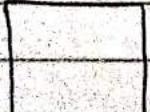
- Balance: Every element inside infographic should balance between each other.
- Unity: This could be about color like if you set primary color is 'red', then you should use shade of red color in the design.

10) What is HTML5 CANVAS? Explain with example.

→ • <canvas> gives you an easy and powerful way to draw graphics. It can be used to draw graphs make photo compositions or do simple animations.

• <html>
<head>
<style> #myCanvas { border: 1px solid black; }
</style>
</head>
<body>
<canvas id="myCanvas" width="100" height="100">
</canvas>
</body>
</html>

• O/P:



Q1

What is SVG? Explain with example.

- SVG stands for "Scalable Vector Graphics".
- SVG is used to define vector-based graphics for the web.
- SVG defines the graphics in XML format.
- Every element and every attribute in SVG files can be animated.
- SVG is a W3C recommendation.
- SVG integrates with other W3C standards such as the DOM and XSL.

Example :-

```
<!DOCTYPE html>
<html>
<body>
<svg width="400" height="100">
  <rect width="300" height="100"
        style="fill: black"/>
</svg>
</body>
</html>
```

O/P :-



Q) What is SVG? Explain with example

- - SVG stands for "Scalable Vector Graphics"
 - SVG is used to define vector-based graphics for the web
 - SVG defines the graphics in XML format.
 - Every element and every attribute in SVG files can be animated
 - SVG is a W3C recommendation
 - SVG integrates with other W3C standards such as the DOM and XSL.

- Example :-

```
<!DOCTYPE html>
<html>
<body>
  <svg width="400" height="100">
    <rect width="300" height="100"
          style="fill: black;" />
  </svg>
</body>
</html>
```

* O/P :-



10. Assignment 2

CO2: Apply core skills for visual analysis.

Module 2

1. List & explain various methods to read data from files with example.
2. What is JSON? Explain with example.
3. List & explain various methods to read data with JSON.
4. List & explain few HTML 5 input types
5. Discuss validation methods with and without HTML5 validation.
6. What is data tables library? How can we use it explain with example?
7. Explain data tables various methods.

* Assignment :- 2

1) List and explain various methods to read data from files.

→ Following methods are there to read data:

1. File Reader, readAs ArrayBuffer()

→ Reads, the content of specified input file.

2. File Reader, readAs BinaryString()

→ Reads the content of the specified I/P file.

→ O/P contains raw binary data from the file as string.

3. File Reader, readAsDataURL()

→ Reads the content of specified I/P file.

→ Result Attribute contains URL ~~not~~ representing file

2) What is JSON? Explain with example.

→ - JSON is an open standard file format.

- JSON stands for JavaScript Object Notation.

- It uses human-readable text to store and transmit data objects.

- Often is used when data sent from server to webpage.

- Example:-

```
var myObj = { name: "John", age: 31, city: "Pune" }
```

```
var myJSON = JSON.stringify(myObj);
```

```
window.location = "demo.json.php?=" + myJSON;
```

3) List and explain various methods to read data with JSON

- The full form of JSON is JavaScript Object Notation
- It is a script file made up of text in programming language
- There are different methods to read JSON file
`json.read()`

Accepts the file object passes the JSON data and returns it.

* `json.loads()`:

- loads can be used to parse JSON string.
- `json.loads()` does not take file path, it takes the file string
- file object.read() as well as `json.loads()` can be used to return the content of the file

`json.loads(json string)`

`json.loads(file object, read())`

4) List and explain HTML5 input type.

- There are various HTML5 input types:
Those are as follows:-

- `<input type = "button">`

- `<input type = "checkbox">`

- `<input type = "email">`

- `<input type = "url">`

- `<input type = "date">`

- `<input type = "text">`

5) Discuss validation methods with and without HTML5 validation.

- - Validation is essential to validate the data entered by user.
- When the data is external, the browser will check to see whether the data is in valid format or not.
- After submission of form, it pop up and are displayed it is server side validation.
- As soon as you complete a field on screen PIPS, it is client side validation.

* Validation Methods:-

1. Native Validation

- It is lightweight as it is implemented at the browser level.
- User don't have to download external scripts.
- Improve performance.

2. Javascript Validation

- It is texture-rich.
- It is robust and more reliable.
- Compatible with older browsers.

* HTML5 Validation:-

<input type="email">

<input type="text" required>

- Q) What is DataTables library? How can we use it?
Explain with example.
- DataTable is a plug-in for the JS library.
 - Highly flexible tool, it adds features to any HTML table.
 - DataTables represent an interesting lowest common denominator in terms of in-browser data visualization.

* Example:

- Use the file that you want to display.
- Input it
- Call the function

```
$(document).ready(function() {  
    $('#Table').DataTable();  
})
```

Q) Explain data table various methods

\$() → Perform JQUERY selection action on the full table.

ajax → Name space for Ajax methods.

ajax.json() → get latest JSON data.

ajax.url() → Get/Set the URL.

data() → Get data for whole table.

`draw()` → Redraw the table

`off()` → Table event removal

`on()` → Table events listener

`cell()` → select a single cell from a table

`search()` → search for data in the table.

`state()` → Get the last saved state of the table.

11. Assignment 3

CO3: Apply visualization techniques for various data analysis tasks.

Module 3

1. List HTML5 Canvas charts techniques. Explain any two with example.
2. List Google API charts techniques. Explain any two with example.

* Assignment :- 3

1) List HTML5 canvas charts techniques. Explain any two with example.

→ Following are various HTML5 canvas techniques:

i. Basic canvas chart

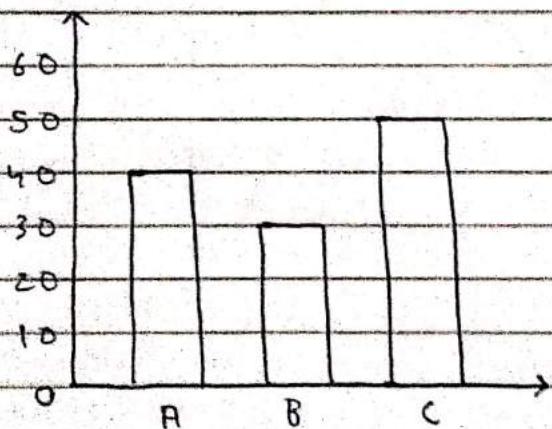
ii. Canvas chart highlight Plot area

iii. Basic canvas chart with guidelines

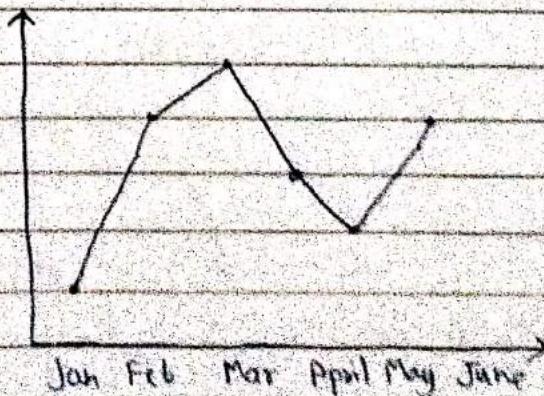
iv. Canvas chart with axis

→ we can draw many types of chart using canvas

e.g. Basic Column Chart:-



* Line chart:-



2) List Google API charts techniques. Explain two with example.

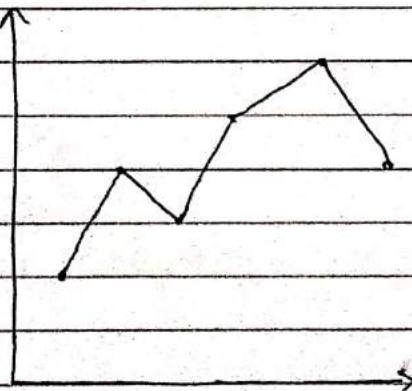
→ Following are Google API chart techniques:-

Barchart Donutchart Linechart Treemap
Pie chart Histogram Bubblechart Columncharts

* Line chart:-

```
var chart = new google.
```

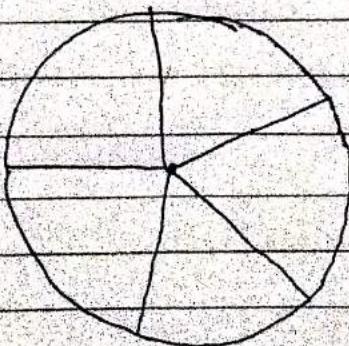
```
visualization.Linechart(document.getElementById  
('chart')); chart.draw(data, options);
```



* Pie chart:-

```
var chart = new google.visualization.
```

```
Piechart(document.getElementById("piechart"));  
chart.draw(data, options);
```



12. Assignment 4

CO4: Design information dashboard.

Module 4

1. How can we use D3 js. Explain with example.
2. List various D3 methods and Explain any two with example.
3. What is animation explain it with example.

* Assignment :- 4

Q)

How can we use D3.js. Explain with example.

- D3 stands for Data Driven Documents.
- D3.js is a javascript library for manipulating documents based on data.
- It is online DO terminology which is dynamic and interactive.
- To use D3, we need:
 - D3.js library → web browser
 - Editor → Web server

→ To use D3.js library include:-

Example:-

```
<html lang="en">
  <head>
    <script src="/Path/d3.min.js">
  </head>
  <body>
    <script>
      // d3 code here
    </script>
  </body>
</html>
```

2)

List various methods. Explain any two.

→ Following are D3.js methods:-

→ append(tag Name):-

Create and append new elements for all the placeholder entries, returning a new selection of created elements.

→ attr(name, fun1):

Transition an attribute value based on data.

→ attr(name):

Get attribute value

→ calc():

Find the first DOM element in the selection.

→ data():

Get the data query for the first group in the selection.

3)

What is animation? Explain with example

→ Animation is a method in which figures are manipulated to appear as moving images.

With the help of animated data visualization, we can engage viewers in ways other methods may not be able to.

- It helps us to convey changes over time.

Example:-

1. Flying bird:

- It is very natural and vivid.
- It makes the display vibrant and engaging.

2. Morphing cube animation:

- Creates gorgeous web animation and content to attract users.

13. Practical 1

CO1: Explain principles of visual perception.

Module 1

Aim: Develop the following Program Using HTML5 CANVAS and SVG TAG

- a. Develop the Different basic Graphical Shapes using HTM5 CANVAS
- b. Develop the Different Advanced Graphical Shapes using HTM5 CANVAS
- c. Develop the Different basic Graphical Shapes using HTM5 SVG
- d. Develop the Different Advanced Graphical Shapes using HTM5 SVG

Code:

Output snapshot: (In output include practical details and execution date & time with your enrollment number)

14. Practical 2

CO1: Explain principles of visual perception.

Module 1

Aim: Develop Following Program Using HTML5 and JavaScript

- a. Develop the simple bar chart using TML5 CANVAS
- b. Read the data .txt file and draw Data Table
- c. Read the data .txt file and draw Simple Bar Chart
- d. Read the data .csv file and draw Data Table
- e. Read the data .csv file and draw Column Bar Chart
- f. Read the data XML file and draw Data Table
- g. Read the data XML file and draw Simple Chart
- h. Read JSON Data and draw Data Table
- i. Read JSON Data and draw Simple Chart

Code:

Output snapshot: (In output include practical details and execution date & time with your enrollment number)

15. Practical 3

CO2: Apply core skills for visual analysis.

Module 3/4

Aim: Develop Following Program Using HTML5 and D3.js and Canvas.js

- a. Showing the data as a column chart (simple)
- b. Showing the data as a stacked column chart
- c. Showing the Data as a column chart for four age group
- d. Showing the data as a Line chart (single, fewer and multiple lines)
- e. Showing the data as a Pie Chart (single and multiple pie)
- f. Showing the data as a Bar Chart (Simple and multiple)

Code:

Output snapshot: (In output include practical details and execution date & time with your enrollment number)

16. Practical 4

CO3: Apply visualization techniques for various data analysis tasks.

Module 5

Aim: Develop Following Program Using HTML5 and Google Charts API and Map API

- a. Using Google Charts API Basics draw charts like a Bar chart
- b. Using Google Charts API Basics draw charts like a Line chart
- c. Using Google Charts API Basics draw PieChart.
- d. Using Google Charts API Basics draw Donut Chart.
- e. Using Google Charts API Basics draw Candle Chart.
- f. Using Google Charts API Basics draw other types of Chart.
- g. Using Google API read JSON file and create Google Map.

Code:

Output snapshot: (In output include practical details and execution date & time with your enrollment number)

17. Practical 5

CO4: Design information dashboard.

Module 6

Aim: Build interconnected Dashboard

Code:

Output snapshot: (In output include practical details and execution date & time with your enrollment number)

Practical 1

CO1: Explain principles of visual perception. Module 1

Aim: Develop the following Program Using HTML5 CANVAS and SVG TAG

- a. Develop the Different basic Graphical Shapes using HTM5 CANVAS
- b. Develop the Different Advanced Graphical Shapes using HTM5 CANVAS
- c. Develop the Different basic Graphical Shapes using HTM5 SVG
- d. Develop the Different Advanced Graphical Shapes using HTM5 SVG

a. Develop the Different basic Graphical Shapes using HTM5 CANVAS

```
<!DOCTYPE html>

<html>

<head>

<title>Practical 1a</title>

</head>

<body onload="draw()" style="font-family: Verdana, sans-serif">

<h4>PRACTICAL-1a</h4>

<h3><b>EARTH R PATEL (200130107522)</b></h3>

<h5 id="time"></h5>

<canvas id="myCanvas" width="1000" height="1000"></canvas>

<script>

function draw() {

var canvas = document.getElementById("myCanvas");

var ctx = canvas.getContext("2d");

ctx.font = "bold 14px Arial";

ctx.fillStyle = "blue";

ctx.fillText("Circle", 450, 125);

ctx.fillStyle = "violet";

ctx.fillText("Triangle", 25, 275);
```

```
ctx.fillStyle = "black";
ctx.fillText("Ellipse", 265, 275);

ctx.fillStyle = "khaki";
ctx.fillText("Line", 450, 275);

ctx.fillText("Square", 40, 125);

ctx.fillStyle = "green";
ctx.fillText("Rectangle", 240, 125);

ctx.fillStyle = "red";
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
currentdate.getMinutes() + ":" + currentdate.getSeconds();
document.getElementById("time").innerHTML = datetime;

ctx.fillRect(10, 10, 100, 100);

ctx.fillStyle = "khaki";
ctx.fillRect(200, 10, 150, 100);

ctx.beginPath();

ctx.fillStyle = "green";
ctx.arc(475, 60, 45, 0, 2 * Math.PI);

ctx.fill();

ctx.beginPath();

ctx.fillStyle = "yellow";
ctx.moveTo(15, 250);
ctx.lineTo(95, 250);
```

```
ctx.lineTo(55, 150);
ctx.closePath(); ctx.fill();
ctx.save();
tx.fillStyle = "blue";
ctx.scale(2, 1);
ctx.beginPath();
ctx.arc(140, 200, 45, 0, 2 * Math.PI);
ctx.fill();
ctx.restore();
ctx.beginPath();
ctx.moveTo(425, 160);
ctx.lineTo(525, 225);
ctx.stroke();
}
</script>
</body>
</html>
```

Output

PRACTICAL-1a

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 15:39:58



Square



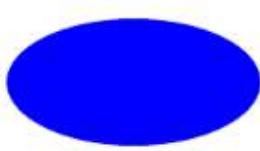
Rectangle



Circle



Triangle



Ellipse



Line

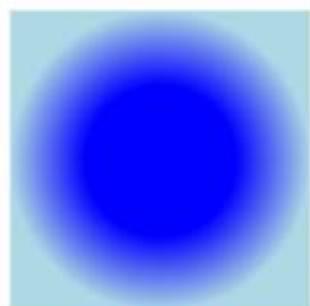
b. Develop the Different Advanced Graphical Shapes using HTM5 CANVAS

```
<!DOCTYPE html>
<html>
<head> <title>PRACTICAL-1b</title> </head>
<body onload="draw()" style="font-family: Verdana, sans-serif">
<h4>PRACTICAL-1b</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<canvas id="myCanvas" width="500" height="500"></canvas>
<script>
function draw() {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
currentdate.getMinutes() + ":" + currentdate.getSeconds();
document.getElementById("time").innerHTML = datetime;
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.font = "italic 14px Verdana";
ctx.fillStyle = "Red";
ctx.fillText("Circular Gradient", 40, 230);
ctx.fillText("Linear Gradient", 300, 250);
ctx.fillText("Image", 70, 400);
ctx.fillText("Arc", 300, 400);
var grd = ctx.createRadialGradient(110, 110, 50, 110, 110, 100);
grd.addColorStop(0, "blue");
grd.addColorStop(1, "lightblue");
ctx.fillStyle = grd;
ctx.fillRect(10, 10, 200, 200);
var grd = ctx.createLinearGradient(250, 0, 450, 0);
grd.addColorStop(0, "blue");
grd.addColorStop(1, "red");
```

```
ctx.fillStyle = grd;  
ctx.fillRect(230, 10, 430, 200);  
var img = new Image();  
img.onload = function () {  
    ctx.drawImage(img, 75, 320);  
};  
img.src = 'https://www.w3schools.com/images/compatible_chrome.png';  
ctx.arc(325, 300, 100, 0, 2 * Math.PI / 4);  
ctx.stroke();  
}  
</script>  
</body>  
</html>
```

Output**PRACTICAL-1b****EARTH R PATEL (200130107522)**

Performed on: 16/4/2022 15:51:7

*Circular Gradient**Linear Gradient**Image**Arc*

c. Develop the Different Advanced Graphical Shapes using HTM5 CANVAS

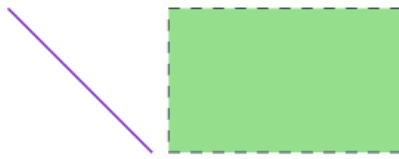
```
<!DOCTYPE html>
<html>
<head> <title>PRACTICAL-1c</title> </head>
<body onload="draw();" style="font-family: Tahoma, Verdana, sans-serif">
<h4>PRACTICAL-1c</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<svg width="700px" height="700px">
<line x1="10" y1="10" x2="140" y2="140" style="stroke: rgb(123, 36, 199); stroke-width: 2"/>
<circle cx="470" cy="85" r="65" style="fill: rgb(54, 91, 210); stroke: rgb(14, 137, 156); stroke-width: 4" />
<rect x="155" y="10" width="210" height="130" style="fill: rgb(112, 211, 102); fill-opacity: 0.75; stroke: rgb(7, 7, 36); stroke-dasharray: 10;" />
<text x="55" y="200">Line</text>
<text x="210" y="200">Rectangle</text>
<text x="470" y="200">Circle</text>
</svg>
<script>
function draw() {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
currentdate.getMinutes(); document.getElementById("time").innerHTML = datetime;
}
</script>
</body>
</html>
```

Output

PRACTICAL-1c

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 15:57



Line



Circle

Rectangle

d. Develop the Different Advanced Graphical Shapes using HTM5 CANVAS

```

<!DOCTYPE html>
<html>
<head> <title>PRACTICAL-1d</title> </head>
<body onload="draw();" style="font-family: Tahoma, Verdana, sans-serif">
<h4>PRACTICAL-1d</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<svg width="1200px" height="500px">
<polygon points="125 10 97 97 10 97 80 150 52 238 125 187 196 238 171 150 240 97 153
97"
fill="plum" style="stroke: rgb(201, 29, 167)" />
<rect x="330" y="75" width="210" height="130" rx="15" ry="15" style="fill: rgb(25, 164, 151);
stroke: rgb(163, 149, 24); stroke-width: 5; opacity: 0.5" />
<line x1="620" y1="140" x2="750" y2="70" style="stroke: rgb(90, 199, 22); stroke-width: 5" />
<line x1="620" y1="140" x2="750" y2="200" style="stroke: rgb(90, 199, 22); stroke-width: 5" />
<line x1="920" y1="10" x2="760" y2="250" style="stroke: black; stroke-width: 5" />
<line x1="1050" y1="140" x2="920" y2="70" style="stroke: rgb(172, 13, 13); stroke-width: 5" />
<line x1="1050" y1="140" x2="920" y2="200" style="stroke: rgb(172, 13, 13); stroke-width: 5" />
<text x="110" y="260" style="fill: rgb(97, 7, 44); font-weight: bold; font-style: italic" >Star
</text>
<text x="370" y="260" style="fill: rgb(151, 146, 2); font-weight: bold; font-style: italic" >
Rounded Rectangle
</text>
<text x="780" y="260" style="fill: rgb(26, 125, 143); font-weight: bold; font-style: italic" >
Code logo
</text>
</svg>
<script>
function draw() {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +

```

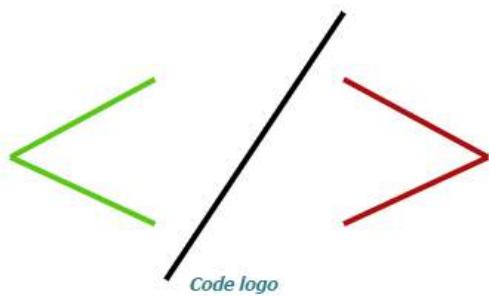
```
(currentdate.getMonth() + 1) + "/" +  
currentdate.getFullYear() + " " +  
currentdate.getHours() + ":" +  
currentdate.getMinutes(); document.getElementById("time").innerHTML = datetime;  
}  
</script>  
</body>  
</html>
```

Output

PRACTICAL-1d

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 16:3



Practical 2

CO1: Explain principles of visual perception. Module 1

Aim: Develop Following Program Using HTML5 and JavaScript

- a. Develop the simple bar chart using HTML5 CANVAS
- b. Read the data .txt file and draw Data Table
- c. Read the data .txt file and draw Simple Bar Chart
- d. Read the data .csv file and draw Data Table
- e. Read the data .csv file and draw Column Bar Chart
- f. Read the data XML file and draw Data Table
- g. Read the data XML file and draw Simple Chart
- h. Read JSON Data and draw Data Table
- i. Read JSON Data and draw Simple Chart

a. Develop the simple bar chart using HTML5 CANVAS

```
<!DOCTYPE html>

<html>

<head> <title>PRACTICAL-2a</title> </head>

<body style="font-family: Verdana, Geneva, Tahoma, sans-serif">

<h4>PRACTICAL-2a</h4>

<h3><b>EARTH R PATEL (200130107522)</b></h3>

<title>Simple Bar Chart</title>

<h5 id="time"></h5>

<div id="chartContainer" style="height: 300px; width: 50%"></div>

<script src="https://canvasjs.com/assets/script/canvasjs.min.js"></script>

<script>

window.onload = function () {

var currentdate = new Date(); var datetime =

"Performed on: " + currentdate.getDate() + "/" +

(currentdate.getMonth() + 1) + "/" +
```

```
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
currentdate.getMinutes() + ":" + currentdate.getSeconds();
document.getElementById("time").innerHTML = datetime;
var chart = new CanvasJS.Chart("chartContainer", {
    title: {
        text: "Simple Bar Chart",
        fontFamily: "Courier",
    },
    axisX: { interval: 1},
    axisY2: {
        interlacedColor: "rgba(238, 130, 238,.2)",
        title: "Students' Grades",
    },
    data: [ {
        type: "bar",
        name: "companies",
        axisYType: "secondary",
        dataPoints: [
            { y: 6, label: "Rajan" },
            { y: 7.5, label: "Jems" },
            { y: 9, label: "Pritesh" },
            { y: 5, label: "Parth" },
            { y: 8, label: "Rocks" },
        ],
    },
},
```

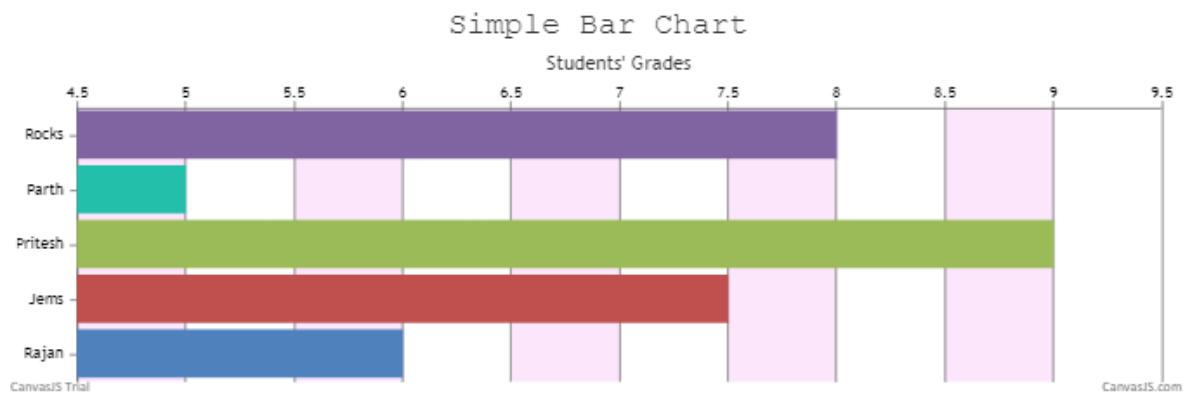
```
],
});  
chart.render();  
};  
</script>  
</body>  
</html>
```

Output

PRACTICAL-2a

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 16:12



b. Read the data .txt file and draw Data Table

```
<!DOCTYPE html>
<html>
<head>
<title>PRACTICAL-2b</title>
</head>
<body style="font-family: Verdana, Geneva, Tahoma, sans-serif">
<h4>PRACTICAL-2b</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<input type="file" name="inputfile" id="inputfile" />
<br />
<pre id="output"></pre>
<table style="width:20%" border="2px">
<tr>
<td>Item</td>
<td>Price/kg</td>
</tr>
<tr>
<td>Lemon</td>
<td>220</td>
</tr>
<tr>
<td>Tomato</td>
<td>40</td>
</tr>
<tr>
<td>Apple</td>
<td>120</td>
</tr>
<tr>
<td>Cherry</td>
<td>350</td>
</tr>
```

```

</table>
<script>
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
currentdate.getMinutes() + ":" + currentdate.getSeconds();
document.getElementById("time").innerHTML = datetime;
};

document.getElementById("inputfile").addEventListener("change", function () {
var fr = new FileReader();
fr.onload = function () {
document.getElementById("output").textContent = fr.result;
};
fr.readAsText(this.files[0]);
});
</script>
</body>
</html>

```

Output

PRACTICAL-2b
EARTH R PATEL (200130107522)

Performed on: 16/4/2022 16:17

No file chosen

Item	Price/kg
Lemon	220
Tomato	40
Apple	120
Cherry	350

c. Read the data .txt file and draw Simple Bar Chart

```
<!DOCTYPE html>

<html>

<head>

<title>PRACTICAL-2c</title>

</head>

<body style="font-family: Verdana, sans-serif">

<h4>PRACTICAL-2c</h4>

<h3><b>EARTH R PATEL (200130107522)</b></h3>

<h5 id="time"></h5>

<input type="file" name="inputfile" id="inputfile" />

<br />

<pre id="output"></pre>

<div id="chartContainer" style="height: 300px; width: 30%"></div>

<script src="https://canvasjs.com/assets/script/canvasjs.min.js"></script>

<script type="text/javascript">

document.getElementById("inputfile").addEventListener("change", function () {

var fr = new FileReader();

fr.onload = function () {

document.getElementById("output").textContent = fr.result;

};

fr.readAsText(this.files[0]);

});

window.onload = function () {

var currentdate = new Date(); var datetime = 

"Performed on: " + currentdate.getDate() + "/" +
```

```
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
currentdate.getMinutes() + ":" + currentdate.getSeconds();
document.getElementById("time").innerHTML = datetime;
var chart = new CanvasJS.Chart("chartContainer", {
    title: { text: "Simple Bar Chart", },
    axisX: { interval: 1},
    axisY2: {
        interlacedColor: "rgba(0,255,255,.3)",
        gridColor: "rgba(1,77,101,.1)",
        title: "Mobile Companies",
    },
    data: [ {
        type: "bar",
        name: "companies",
        axisYType: "secondary",
        color: "#FF6700",
        dataPoints: [
            { y: 20, label: "Redmi" },
            { y: 10, label: "Oneplus" },
            { y: 15, label: "Realme" },
            { y: 26, label: "Apple" },
        ],
    },
],
```

```
});  
chart.render();  
};  
</script>  
</body>  
</html>
```

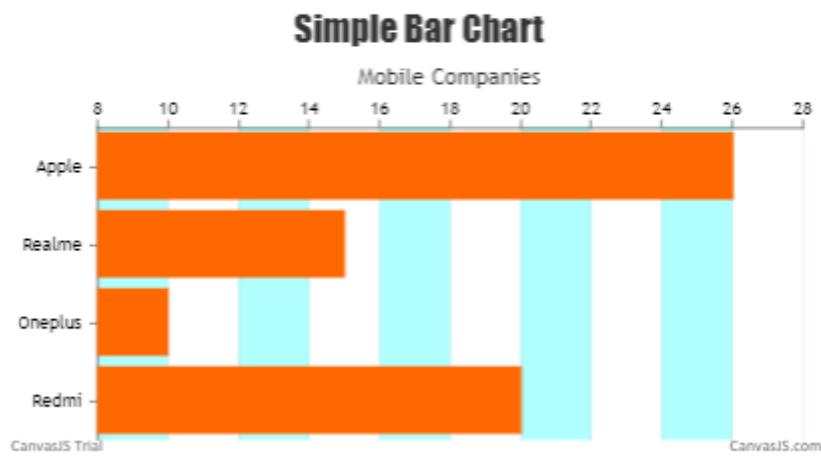
Output

PRACTICAL-2c

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 16:23

No file chosen



d. Read the data .csv file and draw Data Table

```
<!DOCTYPE html>

<html>

<head>

<title>PRACTICAL-2d</title>

</head>

<body style="font-family: Verdana, sans-serif">

<h4>PRACTICAL-2d</h4>

<h3><b>EARTH R PATEL (200130107522)</b></h3>

<h5 id="time"></h5>

<input type="file" id="fileUpload" />

<input type="button" id="upload" value="Upload" onclick="Upload()" />

<div id="dvCSV"></div>

<table style="width: 30%">

<tr>

<td>Name</td>

<td>Number</td>

</tr>

<tr>

<td>Priti</td>

<td>15</td>

</tr>

<tr>

<td>Bholo</td>

<td>25</td>

</tr>
```

```

<tr>
<td>Rajjo</td>
<td>30</td>
</tr>
</table>

<script type="text/javascript">

function Upload() {

var fileUpload = document.getElementById("fileUpload");

var regex = /^[a-zA-Z0-9\s\\.\-]+(\.csv|txt)$/;

if (regex.test(fileUpload.value.toLowerCase())) {

if (typeof FileReader != "undefined") {

var reader = new FileReader();

reader.onload = function (e) {

var table = document.createElement("table");

var rows = e.target.result.split("\n");

for (var i = 0; i < rows.length; i++) {

var cells = rows[i].split(",");
if (cells.length > 1) {

var row = table.insertRow(-1);

for (var j = 0; j < cells.length; j++) {

var cell = row.insertCell(-1);

cell.innerHTML = cells[j];

}

}

}

}

var dvCSV = document.getElementById("dvCSV");

```

```
dvCSV.innerHTML = "";
dvCSV.appendChild(table);
};

reader.readAsText(fileUpload.files[0]);
}

else {

alert("This browser does not support HTML5.");
}

}

else {

alert("Please upload a valid CSV file.");
}

}

var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
currentdate.getMinutes() + ":" + currentdate.getSeconds();
document.getElementById("time").innerHTML = datetime;
</script>

</body>

</html>
```

Output**PRACTICAL-2d****EARTH R PATEL (200130107522)**

Performed on: 16/4/2022 16:34

 marks.csv

Name Marks

Rocks 98

Om 95

Dhruv 96

Het 97

Name Number

Priti 15

Bholo 25

Rajjo 30

e. Read the data .csv file and draw Column Bar Chart.

```
};

reader.readAsText(fileUpload.files[0]);

} else {

alert("This browser does not support HTML5.");

}

} else {

alert("Please upload a valid CSV file.");

}

}

window.onload = function () {

var currentdate = new Date(); var datetime =

"Performed on: " + currentdate.getDate() + "/" +

(currentdate.getMonth() + 1) + "/" +

currentdate.getFullYear() + " " +

currentdate.getHours() + ":" +

currentdate.getMinutes() + ":" + currentdate.getSeconds();

document.getElementById("time").innerHTML = datetime;

var chart = new CanvasJS.Chart("chartContainer", {

title: {

text: "Column Chart ",

},

axisX: { interval: 1},

axisY: {

interlacedColor: "rgba(0, 128, 128,.1)",

gridColor: "crimson",

title: "Fruits",

},

data: [ {

type: "column",

name: "Fruits",

axisYType: "secondary",

color: "blue",

dataPoints: [

{ y: 15, label: "Plum" },
```

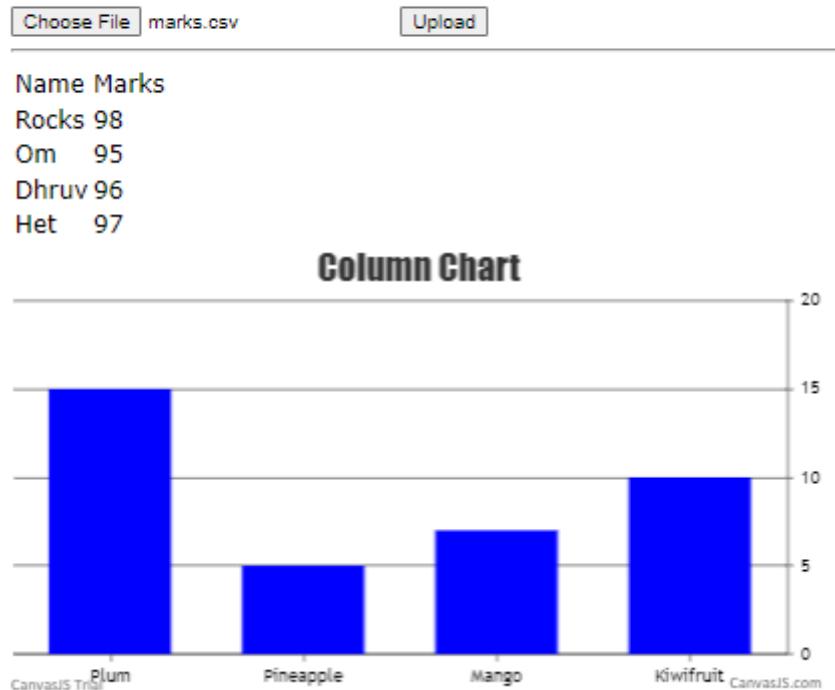
```
{ y: 5, label: "Pineapple" },  
{ y: 7, label: "Mango" },  
{ y: 10, label: "Kiwifruit" },  
],  
}, ],  
});  
chart.render();  
};  
</script>  
</body>  
</html>
```

Output

PRACTICAL-2e

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 16:47



f. Read the data XML file and draw Data Table.

```
<!DOCTYPE html>
<html>
<head>
<title>PRACTICAL-2f</title>
<style>
table {
border-collapse: collapse;
width: 100%;
}
th,
td {
text-align: left;
padding: 8px;
}
tr:nth-child(even) {
background-color: #7ce2af
}
th {
background-color: #7c0f65;
color: white;
}
.button {
position: relative;
text-align: center;
padding: 20px;
border: 4px solid rgb(55, 12, 211);
background: rgba(20, 192, 4, 0.5);
color: rgb(230, 36, 78);
outline: none;
border-radius: 30px;
font-size: 30px;
width: 500px;
}

```

```
.button:hover {  
color: black;  
background: white;  
}  
</style>  
</head>  
<body>  
<h4>PRACTICAL-2f</h4>  
<h3><b>EARTH R PATEL (200130107522)</b></h3>  
<h5 id="time"></h5>  
<center>  
<button type="button" class="button"  
onclick="loadXMLDoc()">  
Get Employees Details  
</button>  
</center>  
<br><br>  
<table id="id"></table>  
<script>  
window.onload = function () {  
var currentdate = new Date(); var datetime =  
"Performed on: " + currentdate.getDate() + "/" +  
(currentdate.getMonth() + 1) + "/" +  
currentdate.getFullYear() + " " +  
currentdate.getHours() + ":" +  
currentdate.getMinutes() + ":" + currentdate.getSeconds();  
document.getElementById("time").innerHTML = datetime;  
}  
function loadXMLDoc() {  
var xmlhttp = new XMLHttpRequest();  
xmlhttp.onreadystatechange = function () {  
// Request finished and response  
// is ready and Status is "OK"  
if (this.readyState == 4 && this.status == 200) {
```

```
empDetails(this);
}
};

xmlhttp.open("GET", "employee.xml", true);
xmlhttp.send();
}

function empDetails(xml) {
var i;
var xmlDoc = xml.responseXML;
var table =
`<tr><th>Firstname</th><th>Lastname</th>
<th>Title</th><th>Division</th>
<th>Building</th><th>Room</th>
</tr>`;
var x = xmlDoc.getElementsByTagName("employee");
for (i = 0; i < x.length; i++) {
table += "<tr><td>" +
x[i].getElementsByTagName("firstname")[0]
.childNodes[0].nodeValue + "</td><td>" +
x[i].getElementsByTagName("lastname")[0]
.childNodes[0].nodeValue + "</td><td>" +
x[i].getElementsByTagName("title")[0]
.childNodes[0].nodeValue + "</td><td>" +
x[i].getElementsByTagName("division")[0]
.childNodes[0].nodeValue + "</td><td>" +
x[i].getElementsByTagName("building")[0]
.childNodes[0].nodeValue + "</td><td>" +
x[i].getElementsByTagName("room")[0]
.childNodes[0].nodeValue + "</td></tr>";
}
document.getElementById("id").innerHTML = table;
}
</script>
</body>
```

</html>

Output**Get Employees Details**

Firstname	Lastname	Title	Division	Building	Room
Rajan	Patel	Engineer	Materials	327	19
Amit	Patel	Accountant	Accts Payable	326	14
Akash	Patel	Engineering Manager	Materials	327	21
Parth	Patel	Engineer	Materials	327	22
Pritesh	Patel	Engineer	Materials	327	24
Vikash	Patel	COO	Management	216	26
Jems	Patel	Accountant	Accts Payable	326	30
Ronak	Patel	COO	Management	216	32
Dhruv	Patel	Engineering Manager	Materials	327	21

g. Read the data XML file and draw Simple Chart.

```
<!DOCTYPE html>
<html>
<head>
<title>PRACTICAL-2g</title>
</head>
<body style="font-family: Verdana, sans-serif">
<h4>PRACTICAL-2g</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id="chartContainer" style="width:50%; height:300px;"></div>
<script type="text/javascript" src="https://canvasjs.com/assets/script/jquery-1.11.1.min.js"></script>
<script type="text/javascript"
src="https://canvasjs.com/assets/script/canvasjs.min.js"></script>
<script type="text/javascript">
window.onload = function() {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
currentdate.getMinutes()+ ":"+ currentdate.getSeconds();
document.getElementById("time").innerHTML = datetime;
var dataPoints = [];
$.get("https://canvasjs.com/services/data/datapoints.php?xstart=5&ystart=10&length=10&type=xml",
function(data) {
$(data).find("point").each(function () {
var $dataPoint = $(this);
var x = $dataPoint.find("x").text();
var y = $dataPoint.find("y").text();
dataPoints.push({x: parseFloat(x), y: parseFloat(y)}));
});
var chart = new CanvasJS.Chart("chartContainer", {
```

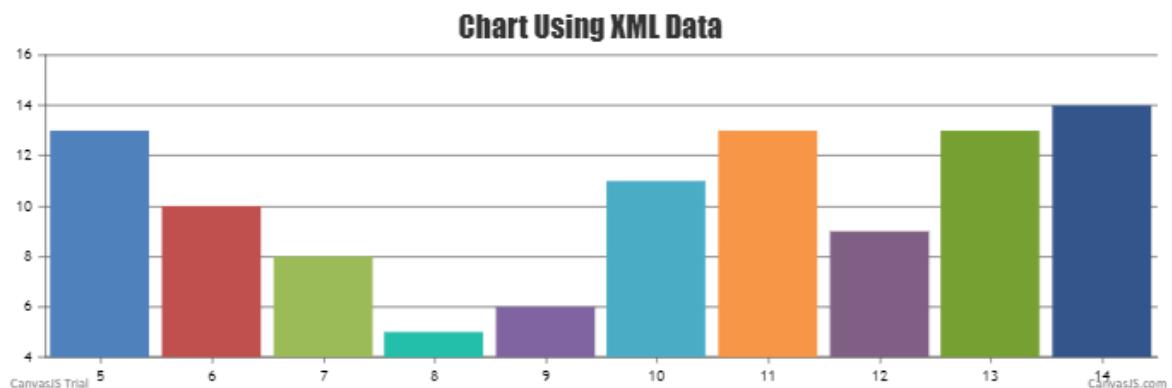
```
title: {  
    text: "Chart Using XML Data",  
},  
data: [{  
    type: "column",  
    dataPoints: dataPoints,  
}]  
});  
chart.render();  
});  
}  
</script>  
</body>  
</html>
```

Output

PRACTICAL-2g

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 17:18



h. Read JSON Data and draw Data Table.

```

<!DOCTYPE HTML>
<html>
<head>
<title>PRACTICAL-2h</title>
</head>
<body style = "text-align:center;">
<h4>PRACTICAL-2h</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<p id = "GFG_UP" style = "font-size: 15px; font-weight: bold;"> </p>
<button onclick = "GFG_FUN()">click here</button>
<br><br>
<table id="table" align = "center" border="1px"></table>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>
<script>
var el_up = document.getElementById("GFG_UP");
var list = [
{"Name":"Rajan", "Roll No.":"7", "Marks":"95"},
 {"Name":"Pritesh", "Roll No.":"30", "Marks":"75"},
 {"Name":"Dhruv", "Roll No.":"25", "Marks":"85"},
 {"Name":"Om", "Roll No.":"37", "Marks":"65"},
];
el_up.innerHTML = "Click on the button to create the "
+ "table from the JSON data.<br><br>"
+ JSON.stringify(list[0]) + "<br>"
+ JSON.stringify(list[1]) + "<br>"
+ JSON.stringify(list[2]) + "<br>"
+ JSON.stringify(list[3]);
function GFG_FUN() {
var cols = [];
for (var i = 0; i < list.length; i++) {
for (var k in list[i]) {
if (cols.indexOf(k) === -1) {

```

```
cols.push(k);
}
}
}

var table = document.createElement("table");
// Create table row tr element of a table
var tr = table.insertRow(-1);
for (var i = 0; i < cols.length; i++) {
var theader = document.createElement("th");
theader.innerHTML = cols[i];
tr.appendChild(theader);
}
for (var i = 0; i < list.length; i++) {
trow = table.insertRow(-1);
for (var j = 0; j < cols.length; j++) {
var cell = trow.insertCell(-1);
cell.innerHTML = list[i][cols[j]];
}
}
var el = document.getElementById("table");
el.innerHTML = "";
el.appendChild(table);
}

var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
currentdate.getMinutes() + ":" + currentdate.getSeconds();
document.getElementById("time").innerHTML = datetime;
</script>
</body>
</html>
```

Output**PRACTICAL-2h****EARTH R PATEL (200130107522)**

Performed on: 16/4/2022 17:20

Click on the button to create the table from the JSON data.

```
{"Name":"Rajan","Roll No.":7,"Marks":95}  
 {"Name":"Pritesh","Roll No.":30,"Marks":75}  
 {"Name":"Dhruv","Roll No.":25,"Marks":85}  
 {"Name":"Om","Roll No.":37,"Marks":65}
```

Name	Roll No.	Marks
Rajan	7	95
Pritesh	30	75
Dhruv	25	85
Om	37	65

i. Read JSON Data and draw a Simple Chart.

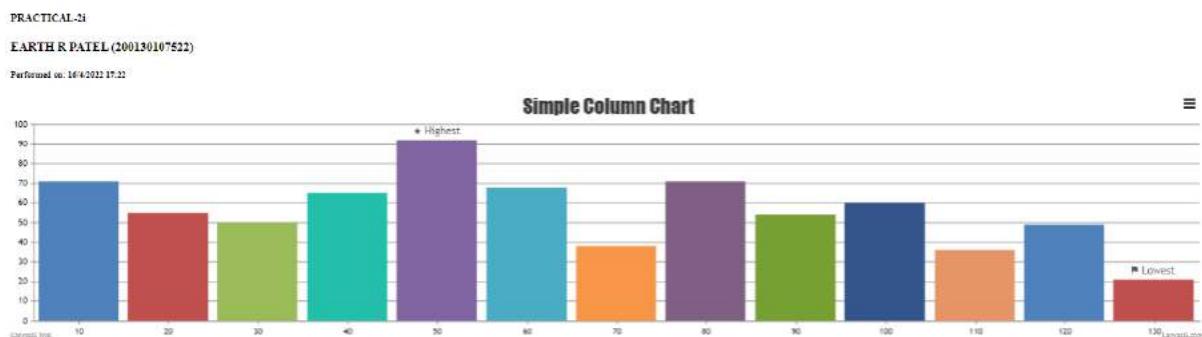
```
<!DOCTYPE html>
<html>
<head>
<title>PRACTICAL-2i</title>
</head>
<body>
<h4>PRACTICAL-2i</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id="chartContainer" style="height: 370px; width: 100%;"></div>
<script src="https://canvasjs.com/assets/script/canvasjs.min.js"></script>
<script>
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
currentdate.getMinutes()+ ":"+ currentdate.getSeconds();
document.getElementById("time").innerHTML = datetime;
var chart = new CanvasJS.Chart("chartContainer", {
animationEnabled: true,
exportEnabled: true,
theme: "light1", // "light1", "light2", "dark1", "dark2"
title:{text: "Simple Column Chart"
},
axisY: {
includeZero: true
},
data: [{type: "column",
indexLabelFontColor: "#5A5757",
```

```

indexLabelFontSize: 16,
indexLabelPlacement: "outside",
dataPoints: [
  { x: 10, y: 71 },
  { x: 20, y: 55 },
  { x: 30, y: 50 },
  { x: 40, y: 65 },
  { x: 50, y: 92, indexLabel: "\u2605 Highest" },
  { x: 60, y: 68 },
  { x: 70, y: 38 },
  { x: 80, y: 71 },
  { x: 90, y: 54 },
  { x: 100, y: 60 },
  { x: 110, y: 36 },
  { x: 120, y: 49 },
  { x: 130, y: 21, indexLabel: "\u2691 Lowest" }
]
}]
});
chart.render();
}
</script>
</body>
</html>

```

Output



Practical 3

CO2: Apply core skills for visual analysis. Module 3/4

Aim: Develop Following Program Using HTML5 and D3.js and Canvas.js

- a. Showing the data as a column chart (simple)
- b. Showing the data as a stacked column chart
- c. Showing the Data as a column chart for four age group
- d. Showing the data as a Line chart (single, fewer and multiple lines)
- e. Showing the data as a Pie Chart (single and multiple pie)
- f. Showing the data as a Bar Chart (Simple and multiple)

a. Showing the data as a column chart (simple)

```
<!DOCTYPE html>

<html>

<head>

<title>PRACTICAL-3a</title>

<style>

.bar {

fill: rgb(4, 95, 95 );

}

</style>

</head>

<body>

<h4>PRACTICAL-3a</h4>

<h3><b>EARTH R PATEL (200130107522)</b></h3>

<h5 id="time"></h5>

<svg width="600" height="500"></svg>

<script src="https://d3js.org/d3.v7.min.js"></script>

<script>
```

```
window.onload = function () {  
    var currentdate = new Date(); var datetime =  
    "Performed on: " + currentdate.getDate() + "/" +  
    (currentdate.getMonth() + 1) + "/" +  
    currentdate.getFullYear() + " " +  
    currentdate.getHours() + ":" +  
    currentdate.getMinutes() + ":" + currentdate.getSeconds();  
    document.getElementById("time").innerHTML = datetime;  
};  
  
var svg = d3.select("svg"),  
margin = 200,  
width = svg.attr("width") - margin,  
height = svg.attr("height") - margin  
svg.append("text")  
.attr("transform", "translate(100,0)")  
.attr("x", 50)  
.attr("y", 50)  
.attr("font-size", "24px")  
.text("XYZ Foods Stock Price")  
  
var xScale = d3.scaleBand().range([0, width]).padding(0.4),  
yScale = d3.scaleLinear().range([height, 0]);  
  
var g = svg.append("g").attr("transform", "translate(" + 100 + "," + 100 + ")");  
  
d3.csv("marks.csv", function(error, data) {  
    if (error) {  
        throw error;  
    }  
})
```

```
xScale.domain(data.map(function(d) { return d.year; }));  
yScale.domain([0, d3.max(data, function(d) { return d.value; })]);  
  
g.append("g")  
  .attr("transform", "translate(0," + height + ")")  
  .call(d3.axisBottom(xScale))  
  .append("text")  
  .attr("y", height - 250)  
  .attr("x", width - 100)  
  .attr("text-anchor", "end")  
  .attr("stroke", "black")  
  .text("Year");  
  
g.append("g")  
  .call(d3.axisLeft(yScale).tickFormat(function(d){  
    return "$" + d;  
  }))  
  .ticks(10))  
  .append("text")  
  .attr("transform", "rotate(-90)")  
  .attr("y", 6)  
  .attr("dy", "-5.1em")  
  .attr("text-anchor", "end")  
  .attr("stroke", "black")  
  .text("Stock Price");  
  
g.selectAll(".bar")  
  .data(data)  
  .enter().append("rect")
```

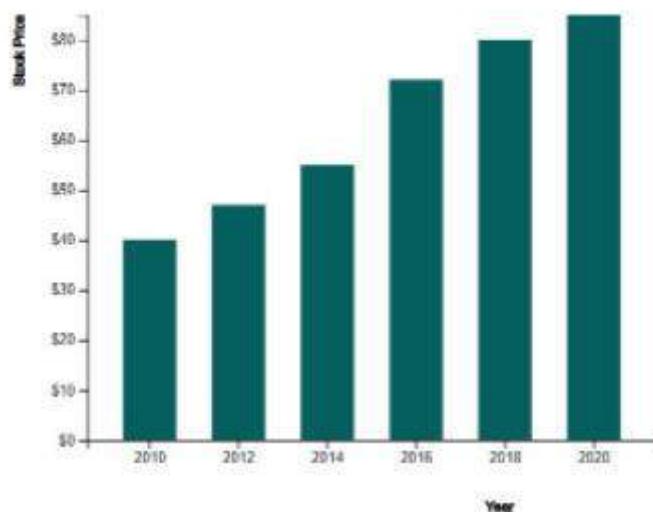
```
.attr("class", "bar")
.attr("x", function(d) { return xScale(d.year); })
.attr("y", function(d) { return yScale(d.value); })
.attr("width", xScale.bandwidth())
.attr("height", function(d) { return height - yScale(d.value); });
});

</script>
</body>
</html>
```

Output**PRACTICAL-3a****EARTH R PATEL (200130107522)**

Performed on: 16/4/2022 17:43

XYZ Foods Stock Price



b. Showing the data as a stacked column chart

```
<!DOCTYPE html>
<html>
<head>
<title>PRACTICAL-3b</title>
<style >
svg {
font: 10px helvetica;
shape-rendering: crispEdges;
}
.axis path,
.axis line {
fill: none;
stroke: #c1c1c1;
}
</style>
</head>
<body>
<h4>PRACTICAL-3b</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<h2>Stacked Bar Chart</h2>
<script src="https://d3js.org/d3.v3.min.js"></script>
<script >
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
};
var margin = 50;
```

```
var width = 600;
height = 300;
var svg = d3.select("body")
.append("svg")
.attr("width", width + margin + 40 )
.attr("height", height + margin + 20)
.append("g")
.attr("transform", "translate(" + (margin+30)/2 + "," + margin/2 + ")");
var data = [
{ year: "2017", alex: "104", mindy: "152", sean: "90", karen: "162" },
{ year: "2018", alex: "122", mindy: "184", sean: "99", karen: "143" },
{ year: "2019", alex: "80", mindy: "201", sean: "127", karen: "114" },
{ year: "2020", alex: "150", mindy: "134", sean: "139", karen: "80" }
];
var colors = ["#FFA500", "#00FF00", "#00FFFF", "#7FFFD4"];
var dataset = d3.layout.stack()(["alex", "mindy", "sean", "karen"].map(function(fruit) {
return data.map(function(d) {
return {x: d3.time.format("%Y").parse(d.year), y: +d[fruit]};
});
}));
var xScale = d3.scale.ordinal()
.domain(dataset[0].map(function(d) { return d.x; }))
.rangeRoundBands([0, width], 0.5);
var yScale = d3.scale.linear()
.domain([0, 600])
.range([height, 0]);
var yAxis = d3.svg.axis()
.scale(yScale)
.orient("left")
.ticks(6)
.tickSize(-width, 0, 0)
.tickFormat( function(d) { return "$" + d } );
var xAxis = d3.svg.axis()
.scale(xScale)
```

```
.orient("bottom")
.tickFormat(d3.time.format("%Y"));
svg.append("g")
.attr("class", "y axis")
.call(yAxis);
svg.append("g")
.attr("class", "x axis")
.attr("transform", "translate(0," + height + ")")
.call(xAxis);
svg.append('text')
.attr('x', width/2)
.attr('y', height + 30)
.attr('text-anchor', 'middle')
.style('font-family', 'Helvetica')
.style('font-size', 12)
.text('Year');
svg.append('text')
.attr('text-anchor', 'middle')
.attr('transform', 'translate(-30,' + height/2 + ')rotate(-90)')
.style('font-family', 'Helvetica')
.style('font-size', 12)
.text('Sale');
var groups = svg.selectAll("g.bars")
.data(dataset)
.enter().append("g")
.attr("class", "bars")
.style("fill", function(d, i) { return colors[i]; })
.style("stroke", "#000");
var rect = groups.selectAll("rect")
.data(function(d) { return d; })
.enter()
.append("rect")
.attr("x", function(d) { return xScale(d.x); })
.attr("y", function(d) { return yScale(d.y0 + d.y); })
```

```
.attr("height", function(d) { return yScale(d.y0) - yScale(d.y0 + d.y); })  
.attr("width", xScale.rangeBand())  
</script>  
</body>  
</html>
```

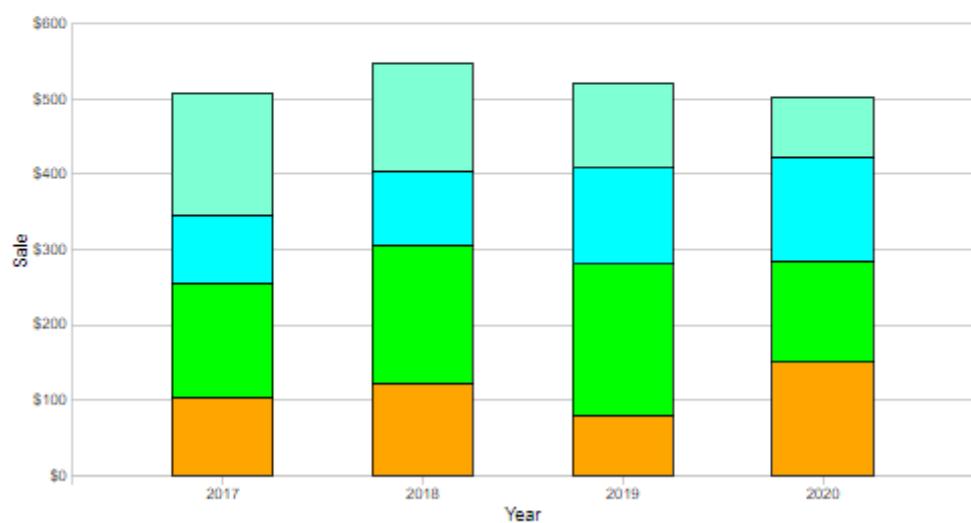
Output

PRACTICAL-3b

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 17:48

Stacked Bar Chart



c. Showing the data as column chart for four age group

```

<!DOCTYPE html>
<html>
<head>
<title>PRACTICAL-3c</title>
<style>
body {
font: 10px sans-serif;
}

.axis path,
.axis line {
fill: none;
stroke: #000;
shape-rendering: crispEdges;
}

.x.axis path {
display: none;
}

</style>
</head>
<body>
<h4>PRACTICAL-3c</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h4 id="time"></h4>
<script src="https://d3js.org/d3.v3.min.js"></script>
<script>
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
}

```

```

};

var margin = {top: 20, right: 20, bottom: 30, left: 40},
width = 600- margin.left - margin.right,
height = 500 - margin.top - margin.bottom;
var x0 = d3.scale.ordinal()
.rangeRoundBands([0, width], .1);
var x1 = d3.scale.ordinal();
var y = d3.scale.linear()
.range([height, 0]);
var xAxis = d3.svg.axis()
.scale(x0)
.tickSize(0)
.orient("bottom");
var yAxis = d3.svg.axis()
.scale(y)
.orient("left");
var color = d3.scale.ordinal()
.range(["#2B65EC", "#EDB32B", "#34A56F", "#A6356C", "#0571b0"]);
var svg = d3.select('body').append("svg")
.attr("width", width + margin.left + margin.right)
.attr("height", height + margin.top + margin.bottom)
.append("g")
.attr("transform", "translate(" + margin.left + "," + margin.top + ")");
d3.json("data.json", function(error, data) {
var categoriesNames = data.map(function(d) { return d.categorie; });
var rateNames = data[0].values.map(function(d) { return d.rate; });
x0.domain(categoriesNames);
x1.domain(rateNames).rangeRoundBands([0, x0.rangeBand()]);
y.domain([0, d3.max(data, function(categorie) { return d3.max(categorie.values, function(d) {
return d.value;});})]);
svg.append("g")
.attr("class", "x axis")
.attr("transform", "translate(0," + height + ")")
.call(xAxis);
});

```

```
svg.append("g")
    .attr("class", "y axis")
    .style('opacity','0')
    .call(yAxis)
    .append("text")
    .attr("transform", "rotate(-90)")
    .attr("y", 6)
    .attr("dy", ".71em")
    .style("text-anchor", "end")
    .style('font-weight','bold')
    .text("Value");

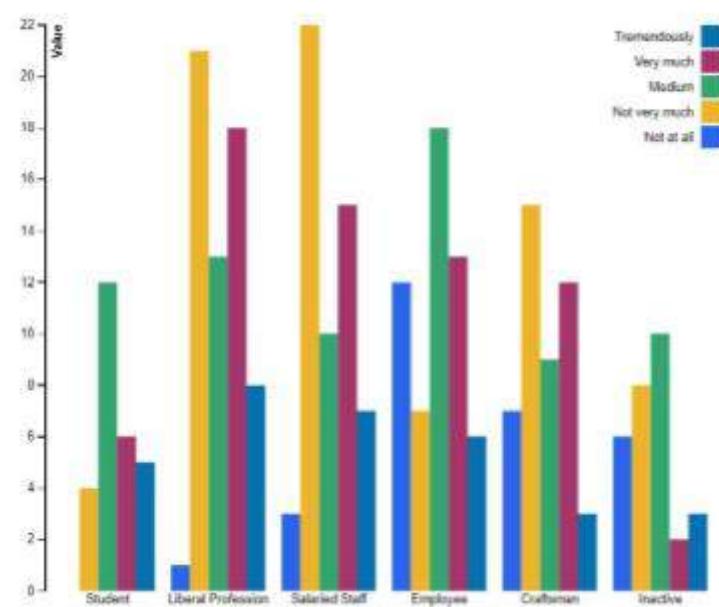
svg.select('.y').transition().duration(500).delay(1300).style('opacity','1');

var slice = svg.selectAll(".slice")
    .data(data)
    .enter().append("g")
    .attr("class", "g")
    .attr("transform",function(d) { return "translate(" + x0(d.categorie) + ",0)"; });
slice.selectAll("rect")
    .data(function(d) { return d.values; })
    .enter().append("rect")
    .attr("width", x1.rangeBand())
    .attr("x", function(d) { return x1(d.rate); })
    .style("fill", function(d) { return color(d.rate) })
    .attr("y", function(d) { return y0(); })
    .attr("height", function(d) { return height - y0(); })
    .on("mouseover", function(d) {
        d3.select(this).style("fill", d3.rgb(color(d.rate)).darker(2));
    })
    .on("mouseout", function(d) {
        d3.select(this).style("fill", color(d.rate));
    });
slice.selectAll("rect")
    .transition()
    .delay(function (d) {return Math.random()*1000;})
```

```
.duration(1000)
.attr("y", function(d) { return y(d.value); })
.attr("height", function(d) { return height - y(d.value); });
var legend = svg.selectAll(".legend")
.data(data[0].values.map(function(d) { return d.rate; }).reverse())
.enter().append("g")
.attr("class", "legend")
.attr("transform", function(d,i) { return "translate(0," + i * 20 + ")"; })
.style("opacity","0");
legend.append("rect")
.attr("x", width - 18)
.attr("width", 18)
.attr("height", 18)
.style("fill", function(d) { return color(d); });
legend.append("text")
.attr("x", width - 24)
.attr("y", 9)
.attr("dy", ".35em")
.style("text-anchor", "end")
.text(function(d) {return d; });
legend.transition().duration(500).delay(function(d,i){ return 1300 + 100 * i;
}).style("opacity", "1");
});
</script>
</body>
</html>
```

Output

PRACTICAL-3c
EARTH R PATEL (200130107522)
Performed on: 16/4/2022 17:52



d. Showing the data as a Line chart (single, fewer and multiple lines)

- Single Line

```
<!DOCTYPE html>
<html>
<head>
<title>PRACTICAL-3d1</title>
</head>
<body>
<h4>PRACTICAL-3d(Single line)</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id="my_dataviz"></div>
<script src="https://d3js.org/d3.v4.js"></script>
<script>
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;};

var svg = d3.select("#my_dataviz")
.append("svg")
.attr("width", width + margin.left + margin.right)
.attr("height", height + margin.top + margin.bottom)
.append("g")
.attr("transform", "translate(" + margin.left + "," + margin.top + ")");
d3.csv("https://raw.githubusercontent.com/holtzy/data_to_viz/master/Example_dataset/3_TwoNumOrdered_comma.csv",
function(d){
return { date : d3.timeParse("%Y-%m-%d")(d.date), value : d.value }};
function(data) {
var x = d3.scaleTime()
.range([ 0, width ]);
```

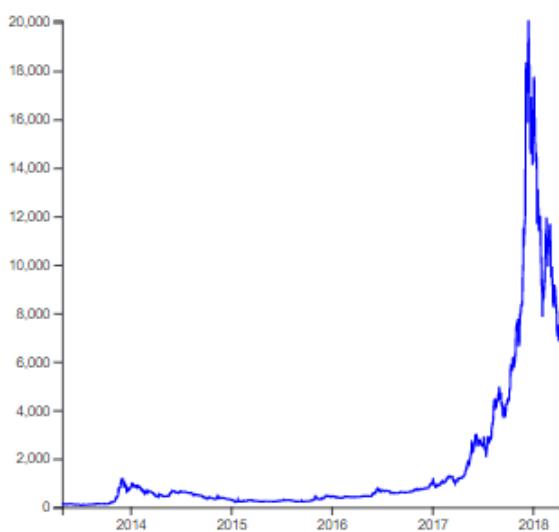
```

svg.append("g")
    .attr("transform", "translate(0," + height + ")")
var y = d3.scaleLinear()
    .domain([0, d3.max(data, function(d) { return +d.value; })])
    .range([-height, 0]);
svg.append("g")
    .call(d3.axisLeft(y));
svg.append("path")
    .datum(data)
    .attr("fill", "none")
    .attr("stroke", "Blue")
    .attr("stroke-width", 1.5)
    .attr("d", d3.line()
        .x(function(d) { return x(d.date) })
        .y(function(d) { return y(d.value) })
    ))
</script>
</body>
</html>

```

Output

PRACTICAL-3d(Single line)
RONAK K PATEL (190130107104)
 Performed on: 16/4/2022 17:55



- **Multiple Line**

```
<!DOCTYPE html>
<html>
<head> <title>PRACTICAL-3d2</title> </head>
<body>
<h4>PRACTICAL-3d(Multiple Lines)</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id="my_dataviz"></div>
<script src="https://d3js.org/d3.v4.js"></script>
<script>
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
};

var margin = {top: 10, right: 30, bottom: 30, left: 60},
width = 460 - margin.left - margin.right,
height = 400 - margin.top - margin.bottom;
var svg = d3.select("#my_dataviz")
.append("svg")
.attr("width", width + margin.left + margin.right)
.attr("height", height + margin.top + margin.bottom)
.append("g")
.attr("transform", "translate(" + margin.left + "," + margin.top + ")");
d3.csv("https://raw.githubusercontent.com/holtzy/data_to_viz/master/Example_dataset/5_On
eCatSevNumOrdered.csv", function(data) {
var sumstat = d3.nest()
.key(function(d) { return d.name;})
.entries(data);
```

```
var x = d3.scaleLinear()
.domain(d3.extent(data, function(d) { return d.year; }))
.range([ 0, width ]);
svg.append("g")
.attr("transform", "translate(0," + height + ")")
.call(d3.axisBottom(x).ticks(5));

var y = d3.scaleLinear()
.domain([0, d3.max(data, function(d) { return +d.n; })])
.range([ height, 0 ]);
svg.append("g")
.call(d3.axisLeft(y));

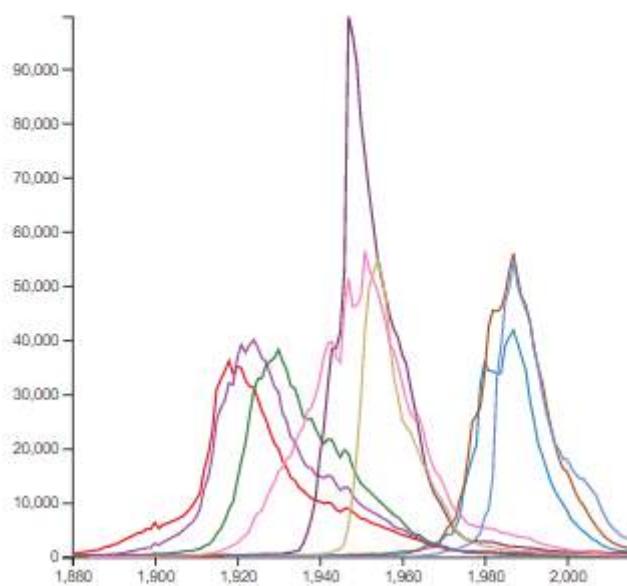
var res = sumstat.map(function(d){ return d.key })
var color = d3.scaleOrdinal()
.domain(res)
.range(['#e41a1c','#368BC1','#387C44','#984ea3','#7D3870','#CFB172','#a65628','#f781bf','#728FCE'])

svg.selectAll(".line")
.data(sumstat)
.enter()
.append("path")
.attr("fill", "none")
.attr("stroke", function(d){ return color(d.key) })
.attr("stroke-width", 1.5)
.attr("d", function(d){
return d3.line()
.x(function(d) { return x(d.year); })
.y(function(d) { return y(+d.n); })
(d.values)
})})

</script>
</body>
</html>
```

Output**PRACTICAL-3d(Multiple Lines)****EARTH R PATEL (200130107522)**

Performed on: 16/4/2022 17:58



e. Showing the data as a Pie Chart (single and multiple pie)

- Single Pie Chart

```
<!DOCTYPE html>
<html>
<head>
<title>PRACTICAL-3e1</title>
<style>
.arc text {
font: 10px sans-serif;
text-anchor: middle;
}
.arc path {
stroke: #fff;
}
.title {
fill: rgb(179, 51, 107);
font-weight: bold;
}
</style>
</head>
<body>
<h4>PRACTICAL-3e(Single Pie Chart)</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<svg width="300" height="350"></svg>
<script src="https://d3js.org/d3.v4.min.js"></script>
<script>
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
```

```
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
};

var svg = d3.select("svg"),
width = svg.attr("width"),
height = svg.attr("height"),
radius = Math.min(width, height) / 2;
var g = svg.append("g")
.attr("transform", "translate(" + width / 2 + "," + height / 2 + ")");
var color = d3.scaleOrdinal(["#e41a1c","#984ea3",'#ff7f00','#4daf4a','#377eb8']);
var pie = d3.pie().value(function(d) {
return d.percent;
});
var path = d3.arc()
.outerRadius(radius - 10)
.innerRadius(0);
var label = d3.arc()
.outerRadius(radius)
.innerRadius(radius - 70);
d3.csv("browser.csv", function(error, data) {
if (error) {
throw error;
}
var arc = g.selectAll(".arc")
.data(pie(data))
.enter().append("g")
.attr("class", "arc");
arc.append("path")
.attr("d", path)
.attr("fill", function(d) { return color(d.data.browser); });
console.log(arc)
arc.append("text")
.attr("transform", function(d) {
return "translate(" + label.centroid(d) + ")";
});
```

```
})
.text(function(d) { return d.data.browser; });
});
svg.append("g")
.attr("transform", "translate(" + (width / 2 - 120) + "," + 20 + ")")
.append("text")
.text("Browser use statistics -Jan 2022")
.attr("class", "title")
</script>
</body>
</html>
```

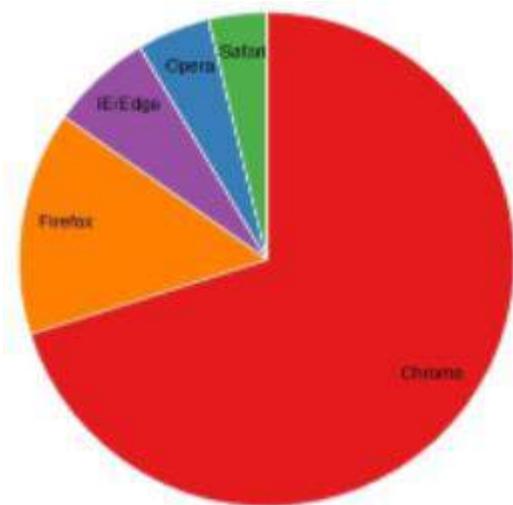
Output

PRACTICAL-3e(Single Pie Chart)

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 18:03

Browser use statistics -Jan 2022



- **Multiple Pie Chart**

```
<html>
<head>
<title>PRACTICAL-3e2</title>
</head>
<body>
<h4>PRACTICAL-3e(�Multiple Pie Chart)</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div class="company-container">
<div class="company-profile"></div>
<div class="company-recommend"></div>
</div>
<div class="company-container">
<div class="company-profile"></div>
<div class="company-recommend"></div>
</div>
<script src="https://d3js.org/d3.v7.min.js"></script>
<script>
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
};
const tickers = [
{
"symbol": "GME",
"buy": 0,
"hold": 3,
"period": "2021-08-01",
```

```
"sell": 5,
"strongBuy": 0,
"strongSell": 2
},
{
"symbol": "AMD",
"buy": 21,
"hold": 16,
"period": "2021-08-01",
"sell": 1,
"strongBuy": 8,
"strongSell": 0
};
const margin = 20;
const width = 250;
const height = 250;
const radius = Math.min(width, height) / 2 - margin;
const data = tickers.map(d => [
{ key: 'Strong Sell', value: d.strongSell },
{ key: 'Sell', value: d.sell },
{ key: 'Hold', value: d.hold },
{ key: 'Buy', value: d.buy },
{ key: 'Strong Buy', value: d.strongBuy },
]);
const color = d3.scaleOrdinal()
.domain(["Strong Sell", "Sell", "Hold", "Buy", "Strong Buy"])
.range(["#307D7E", "#FFFF33", "#931314", "#139494", "#00570c"]);
const pie = d3.pie().value(d => d.value);
const arc = d3.arc()
.innerRadius(0)
.outerRadius(radius);
const recommendSvg = d3.selectAll('.company-recommend')
.data(data)
```

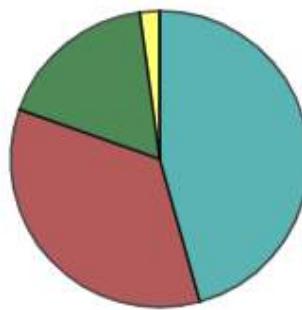
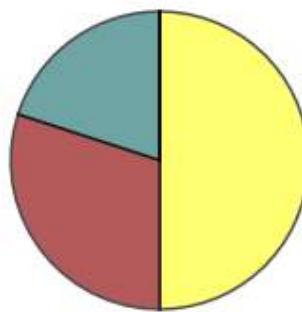
```
.append('svg')
    .attr('width', width)
    .attr('height', height)
    .append('g')
    .attr('transform', `translate(${width / 2},${height / 2})`);
recommendSvg.selectAll('path')
    .data(d => pie(d))
    .join('path')
    .attr('d', arc)
    .attr('fill', d => color(d.data.key))
    .attr('stroke', 'black')
    .attr('stroke-width', '2px')
    .attr('opacity', 0.7);
</script>
</body>
</html>
```

Output

PRACTICAL-3e(Multiple Pie Chart)

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 18:07



f. Showing the data as a Bar Chart (Simple and multiple)**• Simple Bar Chart**

```
<!DOCTYPE HTML>

<html>

<head>

<title>PRACTICAL-3f1</title>

</head>

<body>

<h4>PRACTICAL-3f(Single Bar Chart)</h4>

<h3><b>EARTH R PATEL (200130107522)</b></h3>

<h5 id="time"></h5>

<div id="chartContainer" style="height: 370px; width: 50%;"></div>

<script src="https://canvasjs.com/assets/script/canvasjs.min.js"></script>

<script>

window.onclick = function () {

var currentdate = new Date(); var datetime =

"Performed on: " + currentdate.getDate() + "/" +

(currentdate.getMonth() + 1) + "/" +

currentdate.getFullYear() + " " +

currentdate.getHours() + ":" +

(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();

document.getElementById("time").innerHTML = datetime;

};

window.onload = function () {

var chart = new CanvasJS.Chart("chartContainer", {

animationEnabled: true,

title:{
```

```
text:"Fortune 500 Companies by Country"
```

```
},
```

```
axisX:{
```

```
interval: 1
```

```
},
```

```
axisY2:{
```

```
interlacedColor: "rgba(195, 253, 184,.2)",
```

```
gridColor: "rgba(249, 219, 36,.1)",
```

```
title: "Number of Companies"
```

```
},
```

```
data: [{
```

```
type: "bar",
```

```
name: "companies",
```

```
axisYType: "secondary",
```

```
color: "#2340FA",
```

```
dataPoints: [
```

```
{ y: 3, label: "Canada" },
```

```
{ y: 7, label: "Switzerland" },
```

```
{ y: 4, label: "Britain" },
```

```
{ y: 9, label: "Spain" },
```

```
{ y: 7, label: "Brazil" },
```

```
{ y: 30, label: "India" },
```

```
{ y: 9, label: "Italy" },
```

```
{ y: 15, label: "Australia" },
```

```
{ y: 14, label: "Sweden" },
```

```
{ y: 10, label: "South Korea" },
```

```

{ y: 20, label: "Netherlands" },
{ y: 25, label: "Taiwan" },
{ y: 18, label: "Russia" }

]
})

});

chart.render();

}

```

</script>

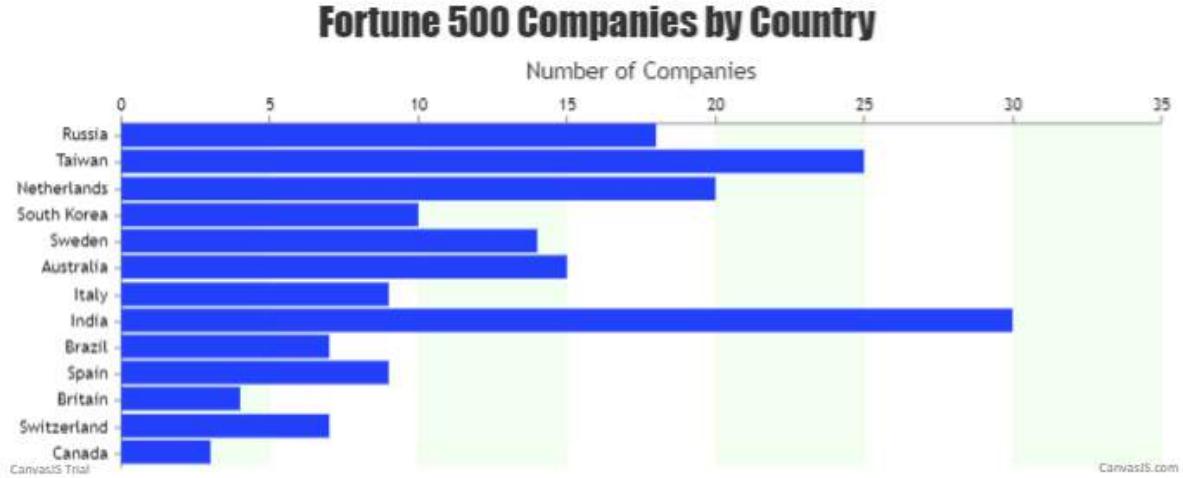
</body>

</html>

Output

PRACTICAL-3f(Single Bar Chart)

EARTH R PATEL (200130107522)



- **Multiple Bar Chart**

```
<!DOCTYPE HTML>
<html>
<head>
<title>PRACTICAL-3f2</title>
</head>
<body>
<h4>PRACTICAL-3f(Multiple Bar Chart)</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id="chartContainer" style="height: 370px; width: 50%;"></div>
<script src="https://canvasjs.com/assets/script/canvasjs.min.js"></script>
<script type="text/javascript">
window.onclick = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
};

window.onload = function () {
var chart = new CanvasJS.Chart("chartContainer", {
animationEnabled: true,
title:{text: "Olympic Medals of all Times (till 2020 Olympics)" },
axisY: {
title: "Medals",
includeZero: true
},
legend: {
cursor:"pointer",

```

```
itemclick : toggleDataSeries
},
toolTip: {
shared: true,
content: toolTipFormatter
},
data: [
{
type: "bar",
showInLegend: true,
name: "Gold",
color: "#FFDF00",
dataPoints: [
{ y: 2500, label: "India" },
{ y: 236, label: "China" },
{ y: 143, label: "France" },
{ y: 276, label: "Great Britain" },
{ y: 354, label: "Germany" },
{ y: 196, label: "Russia" },
{ y: 1100, label: "USA" }
]
},
{
type: "bar",
showInLegend: true,
name: "Silver",
color: "#838996",
dataPoints: [
{ y: 999, label: "India" },
{ y: 186, label: "China" },
{ y: 355, label: "France" },
{ y: 209, label: "Great Britain" },
{ y: 270, label: "Germany" },
{ y: 165, label: "Russia" },
{ y: 806, label: "USA" }
]
```

```
]
},
{
  type: "bar",
  showInLegend: true,
  name: "Bronze",
  color: "#B87333",
  dataPoints: [
    { y: 700, label: "India" },
    { y: 175, label: "China" },
    { y: 500, label: "France" },
    { y: 302, label: "Great Britain" },
    { y: 250, label: "Germany" },
    { y: 188, label: "Russia" },
    { y: 200, label: "USA" }
  ]
})
});

chart.render();

function toolTipFormatter(e) {
  var str = "";
  var total = 0 ;
  var str3;
  var str2 ;
  for (var i = 0; i < e.entries.length; i++){
    var str1 = "<span style= \"color:" +e.entries[i].dataSeries.color + "\">>" +
    e.entries[i].dataSeries.name + "</span>: <strong>" + e.entries[i].dataPoint.y + "</strong>
    <br/>" ;
    total = e.entries[i].dataPoint.y + total;
    str = str.concat(str1);
  }
  str2 = "<strong>" + e.entries[0].dataPoint.label + "</strong> <br/>";
  str3 = "<span style = \"color:Tomato\>Total: </span><strong>" + total + "</strong><br/>";
  return (str2.concat(str)).concat(str3);
```

```

}

function toggleDataSeries(e) {
  if (typeof (e.dataSeries.visible) === "undefined" || e.dataSeries.visible) {
    e.dataSeries.visible = false;
  }
  else {
    e.dataSeries.visible = true;
  }
  chart.render();
}
}

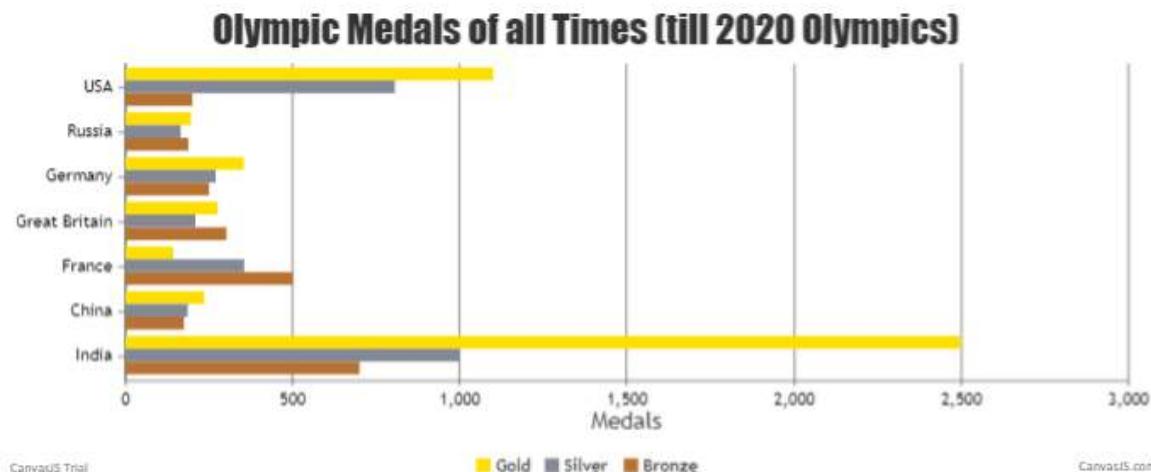
</script>
</body>
</html>

```

Output

PRACTICAL-3d(Multiple Bar Chart)

EARTH R PATEL (200130107522)



Practical 4

CO3: Apply visualization techniques for various data analysis tasks. Module 5

Aim: Develop Following Program Using HTML5 and Google Charts API and Map API

- a. Using Google Charts API Basics draw charts like a Bar chart
- b. Using Google Charts API Basics draw charts like a Line chart
- c. Using Google Charts API Basics draw PieChart.
- d. Using Google Charts API Basics draw Donut Chart.
- e. Using Google Charts API Basics draw Candle Chart.
- f. Using Google Charts API Basics draw other types of Chart.
- g. Using Google API read JSON file and create Google Map.

a. Using Google Charts API Basics draw charts like a Bar chart

```
<html>
<head>
<title>PRACTICAL-4a</title>
</head>
<body>
<h4>PRACTICAL-4a</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id = "container" style = "width: 550px; height: 400px;"> </div>
<script type = "text/javascript" src = "https://www.gstatic.com/charts/loader.js"></script>
<script language = "JavaScript">
google.charts.load('current', {packages: ['corechart']});
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +

```

```

currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();

function drawChart() {

var data = google.visualization.arrayToDataTable([
['Year', 'USA'],
['2016', 350],
['2017', 370],
['2018', 400],
['2019', 450]]);

var options = {title: 'Population (in millions)'};

var chart = new google.visualization.BarChart(document.getElementById('container'));

chart.draw(data, options);

google.charts.setOnLoadCallback(drawChart);

</script>

</body>

</html>

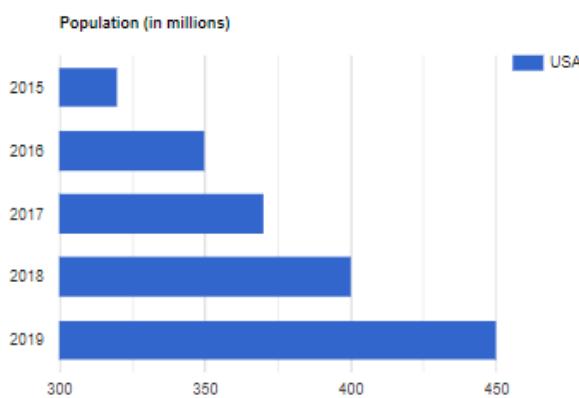
```

Output

PRACTICAL-4a

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 18:21



b. Using Google Charts API Basics draw charts like a Line chart.

```

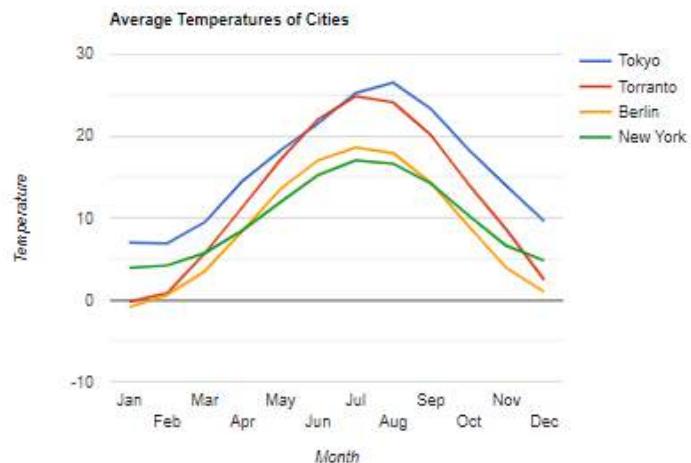
<html>
<head>
<title>PRACTICAL-4b</title>
</head>
<body>
<h4>PRACTICAL-4b</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id = "container" style = "width: 550px; height: 400px; margin:left"></div>
<script src = "https://www.gstatic.com/charts/loader.js"></script>
<script language = "JavaScript">
google.charts.load('current', {packages: ['corechart','line']});
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
};
function drawChart() {
var data = new google.visualization.DataTable();
data.addColumn('string', 'Month');
data.addColumn('number', 'Tokyo');
data.addColumn('number', 'Toronto');
data.addColumn('number', 'Berlin');
data.addColumn('number', 'New York');
data.addRows([
['Jan', 7.0, -0.2, -0.9, 3.9],
['Feb', 6.9, 0.8, 0.6, 4.2],
['Mar', 9.5, 5.7, 3.5, 5.7],
['Apr', 14.5, 11.3, 8.4, 8.5],
]);
}
drawChart();
</script>

```

```
[May', 18.2, 17.0, 13.5, 11.9],  
[Jun', 21.5, 22.0, 17.0, 15.2],  
[Jul', 25.2, 24.8, 18.6, 17.0],  
[Aug', 26.5, 24.1, 17.9, 16.6],  
[Sep', 23.3, 20.1, 14.3, 14.2],  
[Oct', 18.3, 14.1, 9.0, 10.3],  
[Nov', 13.9, 8.6, 3.9, 6.6],  
[Dec', 9.6, 2.5, 1.0, 4.8]  
]);  
  
var options = {'title' : 'Average Temperatures of Cities',  
hAxis: {  
title: 'Month'  
},  
vAxis: {  
title: 'Temperature'  
},  
'width':550,  
'height':400  
};  
  
var chart = new google.visualization.LineChart(document.getElementById('container'));  
chart.draw(data, options);  
}  
google.charts.setOnLoadCallback(drawChart);  
</script>  
</body>  
</html>
```

Output**PRACTICAL-4b****EARTH R PATEL (200130107522)**

Performed on: 16/4/2022 18:23



c. Using Google Charts API Basics draw PieChart.

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>PRACTICAL-4c</title>
</head>
<body>
<h4>PRACTICAL-4c</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id="pie_chart"></div>
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
google.charts.load('current', {"packages": ['corechart']});
google.charts.setOnLoadCallback(drawChart);

function drawChart() {
let data = new google.visualization.DataTable(); data.addColumn('string', 'Label');
data.addColumn('number', 'Part'); data.addRows([
['work', 30],
['watching TV', 10],
['Play', 10],
['sleep', 5],
['eat', 5],
['other',15],
]);
let options = {
"title": 'My Daily Activity', "width": 500,
"height": 500,
}
let chart = new google.visualization.PieChart(document.getElementById('pie_chart'));
chart.draw(data, options);
}

window.onload = function () {
var currentdate = new Date(); var datetime =
```

```
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
};

</script>
</body>
</html>
```

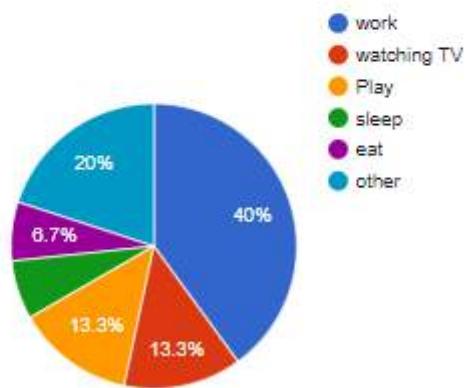
Output

PRACTICAL-4c

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 18:25

My Daily Activity



d. Using Google Charts API Basics draw Donut Chart.

```

<html>
<head>
<title>PRACTICAL-4d</title>
</head>
<body>
<h4>PRACTICAL-4d</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id="donutchart" style="width: 900px; height: 500px;"></div>
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
};
google.charts.load("current", {packages:["corechart"]});
google.charts.setOnLoadCallback(drawChart);
function drawChart() {
var data = google.visualization.arrayToDataTable([
['Task', 'Hours per Day'],
['Work',    10],
['Eat',     2],
['Commute', 2],
['Watch TV', 2],
['Sleep',   6],
['Other',   2]
]);
var options = {

```

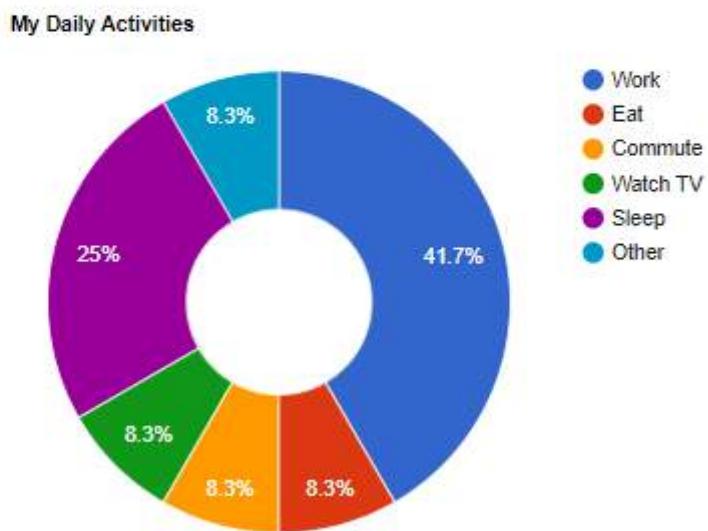
```
title: 'My Daily Activities',
pieHole: 0.4,
};

var chart = new google.visualization.PieChart(document.getElementById('donutchart'));
chart.draw(data, options);
}

</script>
</body>
</html>
```

Output**PRACTICAL-4d****EARTH R PATEL (200130107522)**

Performed on: 16/4/2022 18:28



e. Using Google Charts API Basics draw Candle Chart.

```
<html>
<head>
<title>PRACTICAL-4e</title>
</head>
<body>
<h4>PRACTICAL-4e</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id="chart_div" style="width: 900px; height: 500px"></div>
<script src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
google.charts.load("current", { packages: ["corechart"] });
google.charts.setOnLoadCallback(drawChart);
function drawChart() {
var data = google.visualization.arrayToDataTable([
["Mon", 31, 35, 58, 70],
["Tue", 5, 16, 40, 45],
["Wed", 65, 70, 55, 50],
["Thu", 45, 55, 85, 90],
["Fri", 55, 66, 25, 15],
],
true
);
var options = { legend: "none", candlestick: {
risingColor: {stroke: '#4CAF50', fill: 'Blue'}, fallingColor: {stroke: '#F44336', fill: 'cyan'}
},
colors: ['red']
};
var chart = new google.visualization.CandlestickChart(
document.getElementById("chart_div")
);
chart.draw(data, options);
}
```

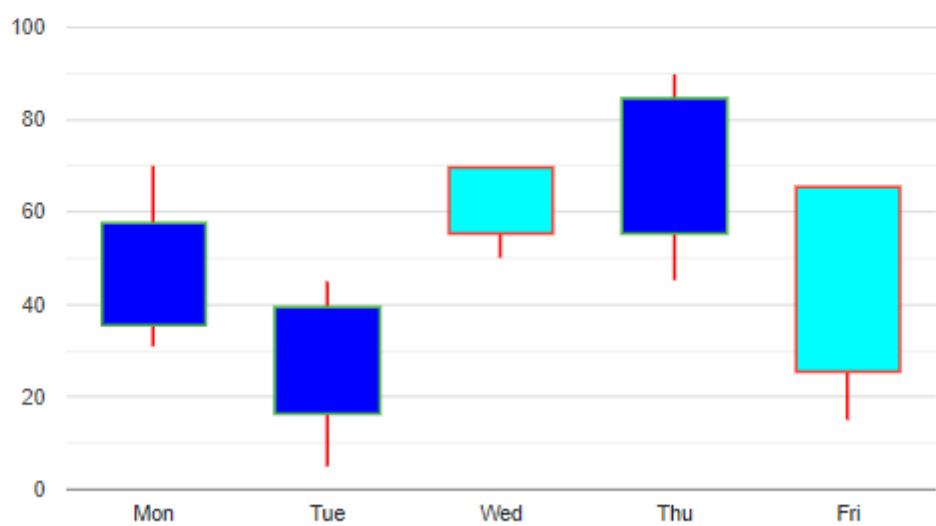
```
window.onload = function () {  
    var currentdate = new Date(); var datetime =  
    "Performed on: " + currentdate.getDate() + "/" +  
    (currentdate.getMonth() + 1) + "/" +  
    currentdate.getFullYear() + " " +  
    currentdate.getHours() + ":" +  
    (currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();  
    document.getElementById("time").innerHTML = datetime;  
};  
</script>  
</body>  
</html>
```

Output

PRACTICAL-4e

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 18:30



f. Using Google Charts API Basics draw other types of Chart.

- **Scatter Chart**

```
<html>
<head>
<title>PRACTICAL-4f1</title>
</head>
<body>
<h4>PRACTICAL-4f(Scatter Chart)</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id="chart_div" style="width: 800px; height: 500px;"></div>
<script type="text/javascript" src="https://www.gstatic.com/charts/loader.js"></script>
<script type="text/javascript">
google.charts.load('current', {'packages':['corechart']});
google.charts.setOnLoadCallback(drawChart);
function drawChart() {
var data = google.visualization.arrayToDataTable([ ['Age', 'Weight'],
[ 2.5,  5],
[ 5,  7.7],
[ 10, 14],
[ 5,  9.9],
[ 4,  8],
[ 7.5, 12.5],
[ 3.5, 7.5],
[ 8, 10],
[ 9, 13.2],
]);
var options = {
title: 'Age vs. Weight comparison',
hAxis: {title: 'Age', minValue: 0, maxValue: 15}, vAxis: {title: 'Weight', minValue: 0, maxValue: 15}, legend: 'none'
};
var chart = new google.visualization.ScatterChart(document.getElementById('chart_div'));
chart.draw(data, options);
}
</script>

```

```

chart.draw(data, options);
}

window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
};

</script>
</body>
</html>

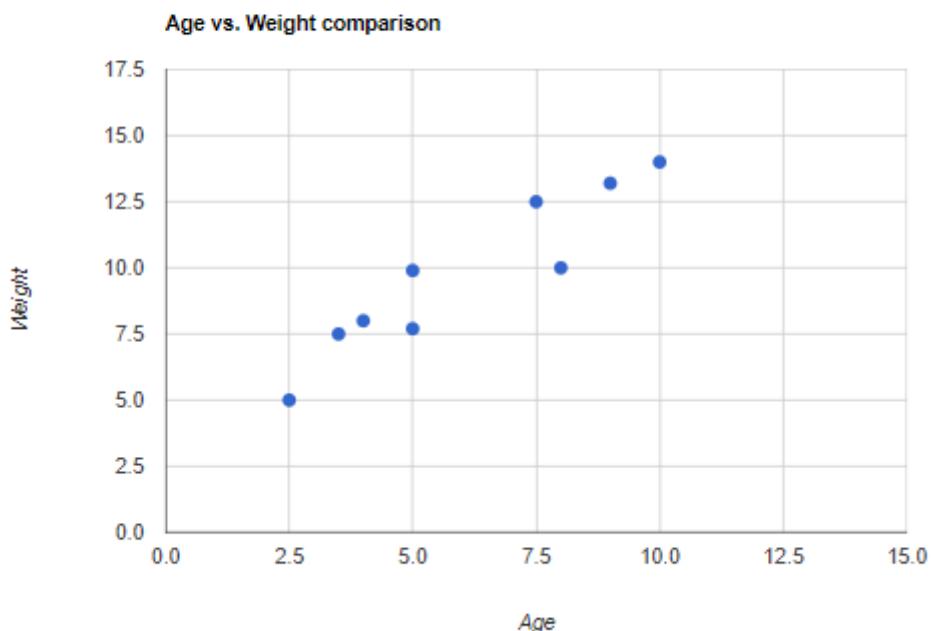
```

Output

PRACTICAL-4f(Scatter Chart)

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 18:32



- **Histogram**

```

<html>
<head>
<title>PRACTICAL-4f2</title>
</head>
<body>
<h4>PRACTICAL-4f(Google Chart)</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<div id = "container" style = "width: 550px; height: 400px;"></div>
<script src="https://www.gstatic.com/charts/loader.js"></script>
<script language = "JavaScript">
window.onload = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
};
google.charts.load('current', {packages: ['corechart']});
function drawChart() {
var data = google.visualization.arrayToDataTable([
['Student Roll No', 'height'],
['1', 70],['2', 45],['3', 68],['4', 60],['5', 49],
['6', 60],['7', 85],['8', 78],['9', 70],['10', 88],
['11', 95],['12', 55],['13', 88],['14', 86],['15', 60],
['16', 84],['17', 65],['18', 62],['19', 85],['20', 75],
['21', 75],['22', 75],['23', 55],['24', 70],['25', 85]
]);
var options = {
title: 'Students height, in cms',
legend: { position: 'none' },

```

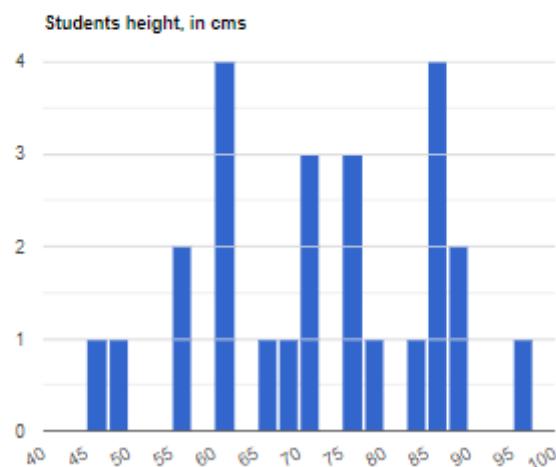
```
histogram: { bucketSize: 4 }  
};  
  
var chart = new google.visualization.Histogram(document.getElementById('container'));  
chart.draw(data, options);  
}  
  
google.charts.setOnLoadCallback(drawChart);  
</script>  
</body>  
</html>
```

Output

PRACTICAL-4f(Google Chart)

EARTH R PATEL (200130107522)

Performed on: 16/4/2022 18:34



g. Using Google API read JSON file and create Google Map.

```
<!DOCTYPE html>
<html>
<head>
<title>PRACTICAL-4g</title>
</head>
<body onload = "loadMap()">
<h4>PRACTICAL-4g</h4>
<h3><b>EARTH R PATEL (200130107522)</b></h3>
<h5 id="time"></h5>
<h2>Ahmedabad Hotels</h2>
<div id = "map" style = "width:640px; height:480px;"></div>
<script>
window.onclick = function () {
var currentdate = new Date(); var datetime =
"Performed on: " + currentdate.getDate() + "/" +
(currentdate.getMonth() + 1) + "/" +
currentdate.getFullYear() + " " +
currentdate.getHours() + ":" +
(currentdate.getMinutes() < 10 ? "0" : "") + currentdate.getMinutes();
document.getElementById("time").innerHTML = datetime;
};

functiongetJSONMarkers() {
const markers = [
{
name: "THE HILLOCK AHMEDABAD",
location: [23.1336, 72.5670]
},
{
name: "Taj Skyline, Ahmedabad",
location: [23.0442, 72.4821]
},
{
name: "Hyatt Regency Ahmedabad",

```

```
location: [23.039287, 72.530601]
}
];
return markers;
}

function loadMap() {
const mapOptions = {
center: new google.maps.LatLng(25.2048, 55.2708),
zoom: 11
}
const map = new google.maps.Map(document.getElementById("map"), mapOptions);

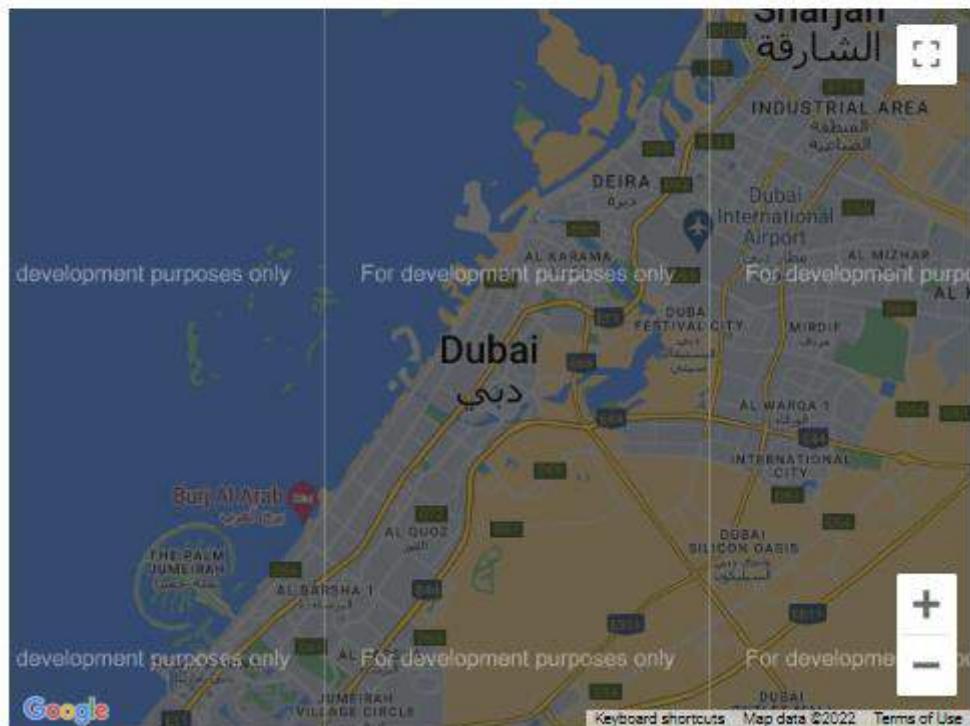
const hotelMarkers = getJSONMarkers();

for(hotel of hotelMarkers) {
let marker = new google.maps.Marker({
map: map,
position: new google.maps.LatLng(hotel.location[0], hotel.location[1]),
title: hotel.name
})
}
}

</script>
<script src = "https://maps.googleapis.com/maps/api/js"></script>
</body>
</html>
```

Output**PRACTICAL-4g****EARTH R PATEL (200130107522)**

Performed on: 16/4/2022 18:37

Ahmedabad Hotels

Practical 5

CO4: Design information dashboard. Module 6

Aim: Build interconnected Dashboard

Output

