Programme	:	**B.Tech.**	Semester	:	**Winter24-25**
Course	:	**BCSE203E:** **Web Programming Lab**	Slot	:	**TE1/TE2**
Faculty	:	**Dr. LM Jenila Livingston**	Marks	:	**10**

Date: 19/02/2025

Exercise 12: JavaScript Canvas, Charts and graphs using plotly.js. and Stack elements using Z-Index

1. Write a **JavaScript program** using the **HTML5 Canvas API** to draw a scene that consists of the following **shapes and corresponding drawings**:

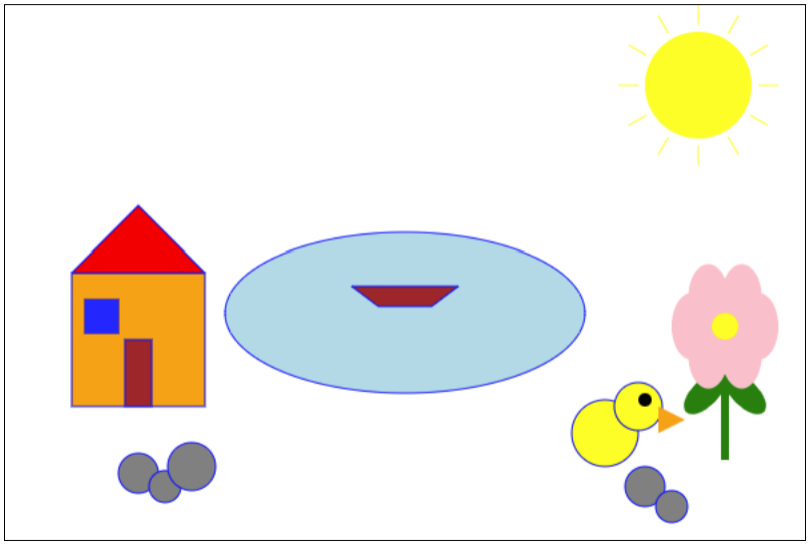
Shape	Drawing Representation
Oval	Pond
Polygon (Quadrilateral with curved edges)	Boat
Two Circles of Different Sizes	Duck (Body & Head)
A Large Circle with Multiple Straight Lines Extending Outward	Sun
A Rectangle with a Triangle on Top	House
An Ellipse with a Vertical Line and Two Curved Shapes	Flower (Stem, Leaves, and Petals)
Multiple Small Circles	Stones

Requirements:

- Use the **Canvas API** functions such as `arc()`, `ellipse()`, `fillRect()`, `lineTo()`, `moveTo()`, and `stroke()`.
- Assign **different colors** to each shape.
- Ensure the **relative positioning** of the elements remains visually structured.

Sample Scene:

Pond Scene using JavaScript Canvas



2. Apply an animation effect to the boat

3. Write a JavaScript program that creates a **working analog clock** using the HTML5 Canvas API. The clock should display the **current time dynamically and accurately**, updating every second.

Requirements:

i) Use the Canvas API to draw the clock face, hands, and markings.
ii) The clock must include the following elements:
    a. A circular clock face with a border and a filled background color.
    b. Hour, minute, and second hands that update dynamically based on the current time.
    c. Numerical or tick markings for hours (1 to 12).
    d. A center pivot point for the hands.
iii) Ensure the hands move smoothly and update every second.

4. Write a **JavaScript program** that dynamically generates the charts (**bar chart, line chart, pie chart** and a **donut chart)** using **Plotly.js**.
   Each chart must include:

    a. Labeled X and Y axes (for bar and line charts).
    b. Title for each chart.
    c. Different colors for data points.
    d. Legend (for the pie chart and donut) showing categories.

ii) The chart should be scaled properly to fit within the display area.

5. Write a JavaScript program that dynamically creates and manipulates **overlapping elements** using **CSS `z-index`**. The program should allow the user to **change the stacking order** of elements by adjusting their `z-index` values.

    - Create at least three overlapping elements (e.g., `div` boxes or images).
    - Use CSS z-index to control the layering order of these elements.
    - Provide buttons or user input to dynamically adjust the z-index values using JavaScript.
    - Display the current z-index value of each element.