

Lahore Garrison

University

DHA Phase 6 Lahore

Lab Task 11



Student Name	Ateeb Qaiser
Student Roll No	Fa23/BSCS/279
Section	G
Subject	Web Technologies Lab
Instructor Name	Mr. M Yousaf

Lab 11: Canvas & SVG — Interactive Graphics

Objectives:

Use Canvas API and SVG to draw and animate simple graphics; build a simple drawing app.

Tools/Tech:

- VS Code, Canvas API, optionally a library (paper.js) — optional

Tasks/Steps:

1. Create a canvas and draw shapes/text/images via JS.
2. Implement basic user drawing (mouse events) and a clear/save as image feature.
3. Create a simple SVG animation (e.g., moving object).

Code:

```
<!DOCTYPE html>
<html>
<head>
<title>Advanced Canvas + SVG Lab</title>

<style>
/* ----- GLOBAL THEME ----- */
body{
    background:#0d0f1a;
    color:#e6f7ff;
    font-family:Trebuchet MS, sans-serif;
    text-align:center;
    padding-bottom:40px;
}

/* Header */
h1{
    text-shadow:0 0 30px #00eaff;
    color:#00f7ff;
}

/* Canvas UI */
#board{
    background:#fff;
    border:4px solid #00f7ff;
    border-radius:12px;
    box-shadow:0 0 30px #00eaff;
    cursor:crosshair;
}

/* Buttons + Inputs */
button, select, input{
```

```

margin:7px;
padding:8px 15px;
font-size:15px;
background:#021a2d;
border:2px solid #00eaff;
color:#d1ffff;
border-radius:6px;
transition:0.2s;
cursor:pointer;
}
button:hover,select:hover,input:hover{
  box-shadow:0 0 15px cyan;
}

/* Color Swatches */
.colors div{
  height:28px; width:28px; border-radius:50%;
  display:inline-block; margin:4px; cursor:pointer;
  border:2px solid white;
}

/* SVG Section Card */
#svgBox{
  margin-top:35px;
  background:#07131f;
  padding:25px;
  border-radius:12px;
  box-shadow:0 0 25px #00eaff;
}
</style>
</head>
<body>

<h1>🎨 Neon Drawing Studio + ○ Animated Ball</h1>

<!-- ====== CANVAS SECTION ===== -->
<!-- Task 1: Create a canvas and draw shapes/text/images via JS -->
<canvas id="board" width="850" height="460"></canvas>
<br><br>

<!-- COLORS -->
<div class="colors">
  <div style="background:black" onclick="setColor('black')"></div>
  <div style="background:red" onclick="setColor('red')"></div>
  <div style="background:blue" onclick="setColor('blue')"></div>
  <div style="background:green" onclick="setColor('green')"></div>
  <div style="background:yellow" onclick="setColor('yellow')"></div>
  <div style="background:purple" onclick="setColor('purple')"></div>
  <div style="background:orange" onclick="setColor('orange')"></div>
</div>

```

```

<!-- USER CUSTOM PICK COLOR -->

<!-- GRADIENT MODE -->
Color Picker
<input type="color" id="pick" onchange="setColor(this.value)">
<br><br>
Brush Size
<input type="range" id="size" min="2" max="50" value="6">

<b><!-- Task 2: Implement basic user drawing (mouse events) and a clear/save as image feature --></b>
<button onclick="eraser()">Eraser</button>
<button onclick="clearCanvas()">Clear</button>
<button onclick="saveImage()">Save PNG</button>

<br><br><hr style="width:70%;border-color:#00eaff"><br>

<b><!-- ====== SVG ANIMATION SECTION ====== --></b>
<div id="svgBox">
<h2>○ Moving Ball Animation (Speed + Color + Gradient)</h2>

<b><!-- Task 3: Create a simple SVG animation (e.g., moving object) --></b>
<svg width="750" height="150">
<defs>
  <linearGradient id="g1" x1="0%" x2="100%">
    <stop offset="0%" stop-color="#00f7ff"/>
    <stop offset="100%" stop-color="#ff00ff"/>
  </linearGradient>
</defs>

<circle id="ball" cx="50" cy="75" r="30" fill="url(#g1)"></circle>
</svg>

<br><br>

<select id="bColor" onchange="ballColor(this.value)">
<option value="cyan">Cyan</option>
<option value="red">Red</option>
<option value="yellow">Yellow</option>
<option value="lime">Green</option>
<option value="gradient">🌈 Gradient</option>
</select>

<button onclick="setSpeed(1)">Slow</button>
<button onclick="setSpeed(3)">Medium</button>
<button onclick="setSpeed(6)">Fast</button>
<button onclick="stopBall()">Stop / Start</button>
</div>

<script>
/* ====== ■ CANVAS DRAWING LOGIC ====== */
```

```

const canvas = document.getElementById("board");
const ctx = canvas.getContext("2d");
let draw = false, color = "black", size = 6, grad = false;

// Mouse events for drawing
canvas.addEventListener("mousedown", () => draw = true);
canvas.addEventListener("mouseup", () => { draw = false; ctx.beginPath(); });
canvas.addEventListener("mousemove", e => {
    if (!draw) return;
    ctx.lineWidth = size;
    ctx.lineCap = "round";
    ctx.strokeStyle = grad ? createGradient(e.offsetX, e.offsetY) : color;
    ctx.lineTo(e.offsetX, e.offsetY);
    ctx.stroke();
    ctx.beginPath();
    ctx.moveTo(e.offsetX, e.offsetY);
});
document.getElementById("size").oninput = e => size = e.target.value;
document.getElementById("pick").oninput = e => color = e.target.value;

// Gradient creation
function createGradient(x, y) {
    let g = ctx.createLinearGradient(0, 0, x, y);
    g.addColorStop(0, "cyan");
    g.addColorStop(1, "magenta");
    return g;
}

// Canvas utility functions
function setColor(c) { color = c; grad = false; }
function toggleGradient() { grad = !grad; }

function eraser() { color = "white"; grad = false; }
function clearCanvas() { ctx.clearRect(0, 0, canvas.width, canvas.height); }

function saveImage() {
    let link = document.createElement('a');
    link.download = "Drawing.png";
    link.href = canvas.toDataURL();
    link.click();
}

/* ====== ○ SVG BALL ANIMATION TOGGLEABLE ====== */
let ball = document.getElementById("ball");
let x = 50, speed = 3, moving = true;

function animate() {
    if (moving) {
        x += speed;
        if (x >= 720 || x <= 30) speed *= -1;
    }
}

```

```

        ball.setAttribute("cx", x);
    }
    requestAnimationFrame(animate);
}
animate();

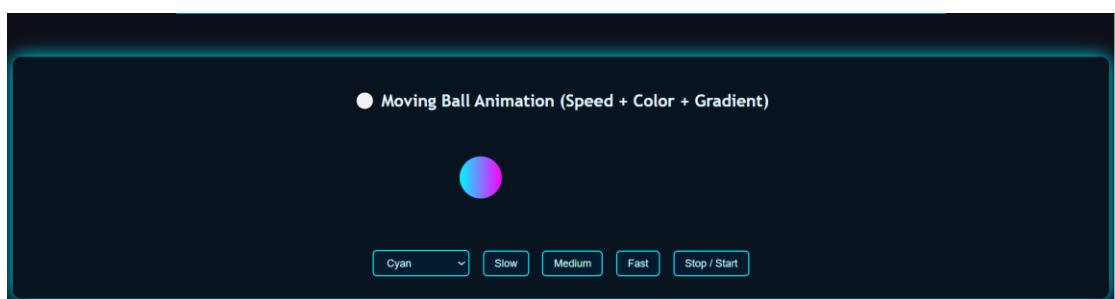
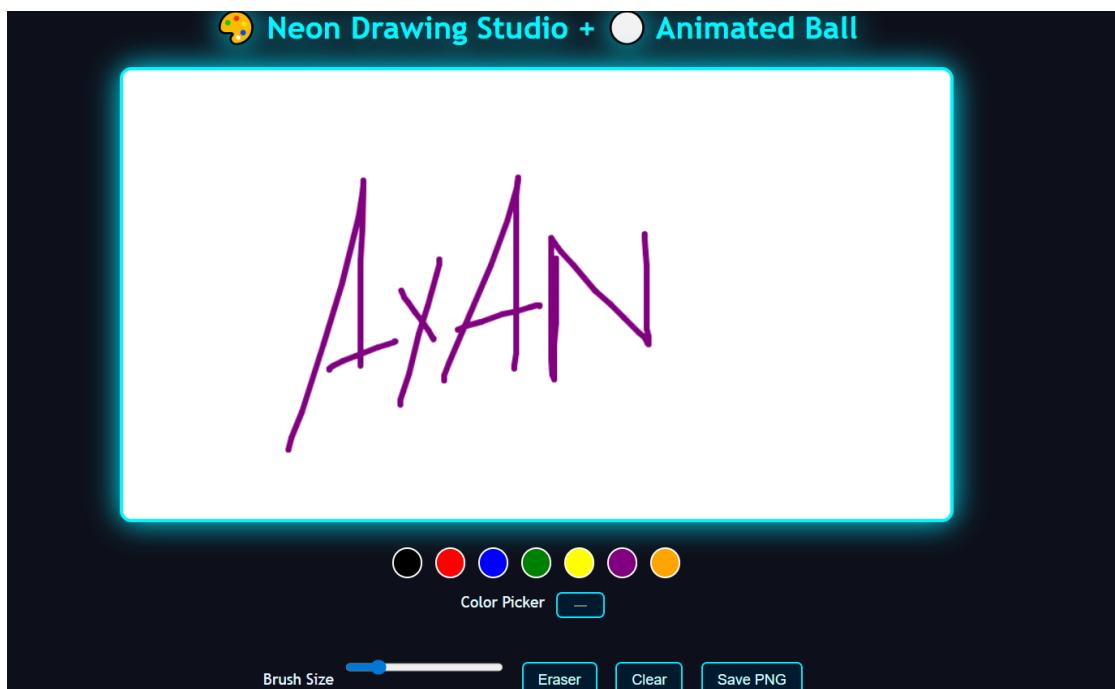
// Control functions for ball animation
function setSpeed(v) { speed = v; }
function stopBall() { moving = !moving; } // toggles stop/start

function ballColor(c) {
    if (c == "gradient") ball.setAttribute("fill", "url(#g1)");
    else ball.setAttribute("fill", c);
}
</script>

</body>
</html>

```

Output:



Deliverables:

Drawing app and an example SVG animation; code and demo.

CONCLUSION:

In this lab, we implemented a dynamic canvas for freehand drawing with color, size, and gradient options. We also created an interactive SVG animation with controllable speed and color, demonstrating basic graphics and animation in web development.

RUBRICS:

Performance			Lab Report		
Description	Total Marks	Marks Obtained	Description	Total Marks	Marks Obtained
Ability to Conduct practical	5		Structure	5	
Data Analysis & Interpretation	5		Efficiency	5	
Total Marks obtained			Total Marks Obtained		

Instructor Signature _____