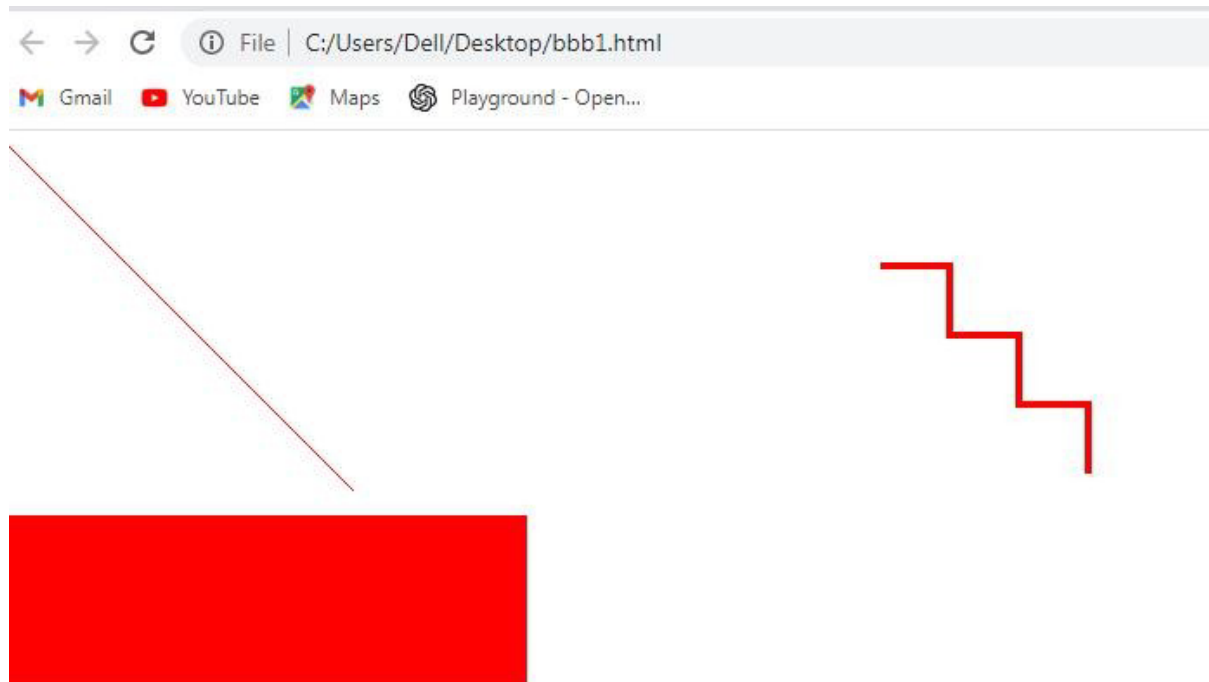


PART B

1. Write an HTML program to draw line, polyline and rectangle and fill rectangle with red color using svg tag

```
<html>
<body>
<svg height="210" width="500">
<line x1="0" y1="0" x2="200" y2="200" style="stroke:rgb(255,0,0);stroke-width:2" /> Sorry,
your browser does not support inline SVG.
</svg>
<svg height="180" width="500">
<polyline points="0,40 40,40 40,80 80,80 80,120 120,120 120,160"
style="fill:white;stroke:red;stroke-width:4" /> Sorry,
your browser does not support inline SVG.
</svg>
<svg width="400" height="110">
<rect width="300" height="100" style="fill:rgb(255,0,0);stroke-width:3;" /> Sorry,
your browser does not support inline SVG.
</svg>
</body>
</html>
```

Output:



2. Write an HTML program to draw star and multiple circle and with different color using svg tag

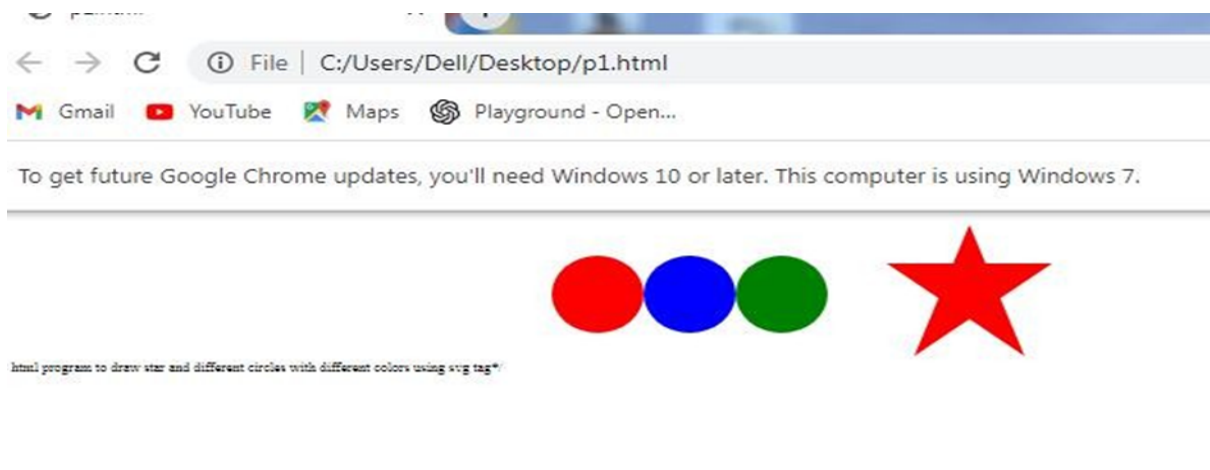
```

<html>
<body>

<svg width="400" height="200">
<circle cx="100" cy="100" r="50" fill="red" />
<circle cx="200" cy="100" r="50" fill="blue" />
<circle cx="300" cy="100" r="50" fill="green" />
</svg>
<svg id = "svgelem" height = "200" xmlns = "http://www.w3.org/2000/svg">
<polygon points = "100,10 40,180 190,60 10,60 160,180" fill = "red"/>
</svg>
</body>
</html>

```

Output:



2B. Write an HTML program to create logo with linear gradient properties using svg tag.

```

<html>
<body>
<svg height="500" width="700">
<defs>
<linearGradient id="grad1" x1="0%" y1="0%" x2="100%" y2="0%">
<stop offset="0%"
style="stop-color:rgb(255,255,0);stop-opacity:1" />
<stop offset="100%"
style="stop-color:rgb(255,0,0);stop-opacity:1" />
</linearGradient>
</defs>
<ellipse cx="100" cy="70" rx="85" ry="55" fill="url(#grad1)" />
<text fill="#ffffff" font-size="45" font-family="Verdana" x="50"
y="86">MIT</text>
</svg>
</body>
</html>

```

Output:



3. Write an HTML program to draw square and rectangle using canvas tag and javascript

```
//Rectangle//
<html>
<body>
<h1>Rectangle</h1>
<canvas id="myCanvas">Your browser does not support the canvas
tag.</canvas>
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.fillStyle = "#FF0000";
ctx.fillRect(0, 0, 180, 80);
</script>
</body>
</html>
```

Output

Rectangle

Rectangle



// Square//

```
<html>
<body>
<h1>The Square canvas element</h1>
<canvas id="myCanvas" onmouseover="c()">Your browser does not support
the canvas tag.</canvas>
<script>
var c = document.getElementById("myCanvas");
var ctx = c.getContext("2d");
ctx.fillStyle="#FF3456";
ctx.fillRect(0, 0, 80, 80);
</script>
</body>
</html>
```

Output

Square

The Square canvas element



4. Write an HTML program to draw bezier curve using canvas tag and JavaScript

```
<!DOCTYPE html>
```

```

<html>

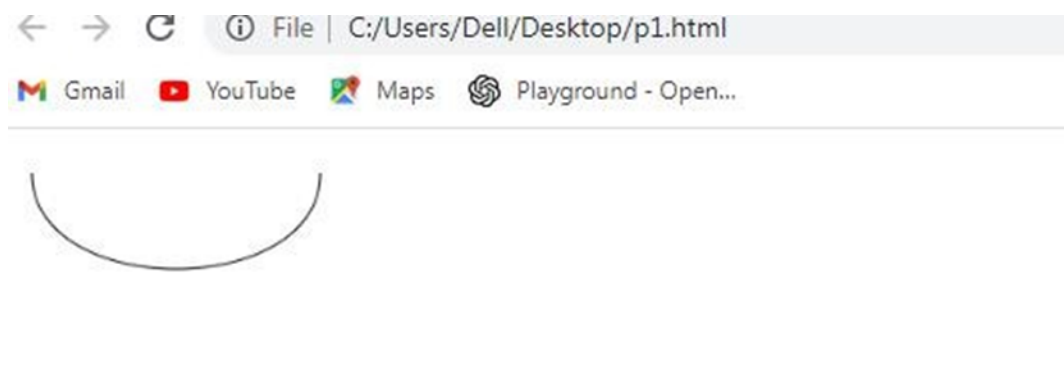
<head>
  <script
    type="text/javascript">
    function draw() {
      var canvas = document.getElementById('canvas');
      if (canvas.getContext) {
        var ctx = canvas.getContext('2d');
        }
        ctx.beginPath();
        ctx.moveTo(20,
          20);
        ctx.bezierCurveTo(20, 100, 200, 100, 200, 20);
        ctx.stroke();
      }
    </script>
  </head>

  <body onload="draw();">
    <canvas id="canvas" width="220" height="220"></canvas>
  </body>

</html>

```

Output:



5. Write an HTML program to import an external image into a canvas and then to draw on that image.

```

<html>

<head>

<title>Draw on Image</title>

```

```
<style>

  canvas {

    border: 1px solid black;

  }

</style>

</head>

<body>

<h2>Draw on Image</h2>

<canvas id="canvas"></canvas>

<script>

  const canvas = document.getElementById('canvas');

  const ctx = canvas.getContext('2d');

  let isDrawing = false;

  let startX, startY;

  const image = new Image();

  image.src = 'https://via.placeholder.com/400'; // Replace with the URL of your image

  image.onload = function() {

    canvas.width = image.width;

    canvas.height = image.height;

    ctx.drawImage(image, 0, 0);

  };

  canvas.addEventListener('mousedown', (e) => {

    isDrawing = true;

    startX = e.offsetX;

    startY = e.offsetY;
```

```
});

canvas.addEventListener('mousemove', (e) => {

    if (!isDrawing) return;

    const x = e.offsetX;

    const y = e.offsetY;

    ctx.beginPath();

    ctx.moveTo(startX, startY);

    ctx.lineTo(x, y);

    ctx.stroke();

    startX = x;

    startY = y;

});

canvas.addEventListener('mouseup', () => isDrawing = false);

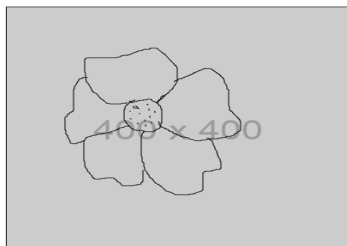
</script>

</body>

</html>
```

Output:

Draw on Image



6. Write an html program to draw rectangle box using canvas and change background to red and change scale of rectangle to 2 on hover properties.

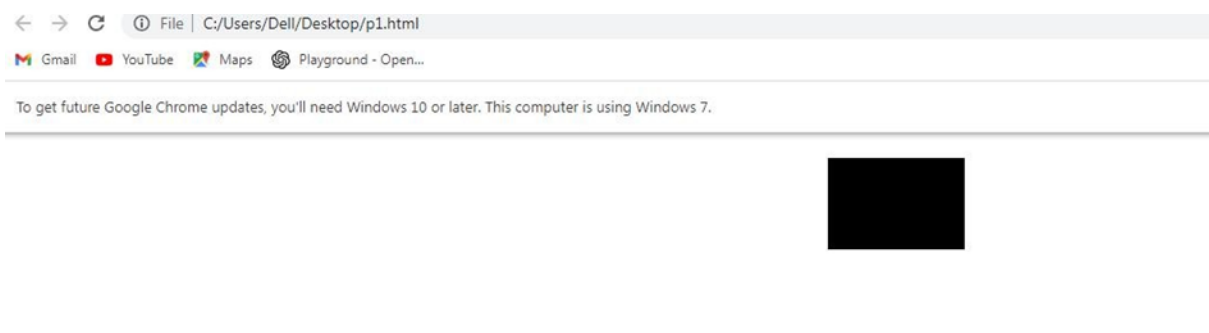
```
<!DOCTYPE html>
<html>
<head>
<script>
function drawRectangle()
{
    var c =
    document.getElementById("myCanvas");
    var ctx = c.getContext("2d");
    ctx.fillStyle = "black";
    ctx.fillRect(20,20,150,100);
    c.addEventListener('mouseover',onMouseO
ver);
}

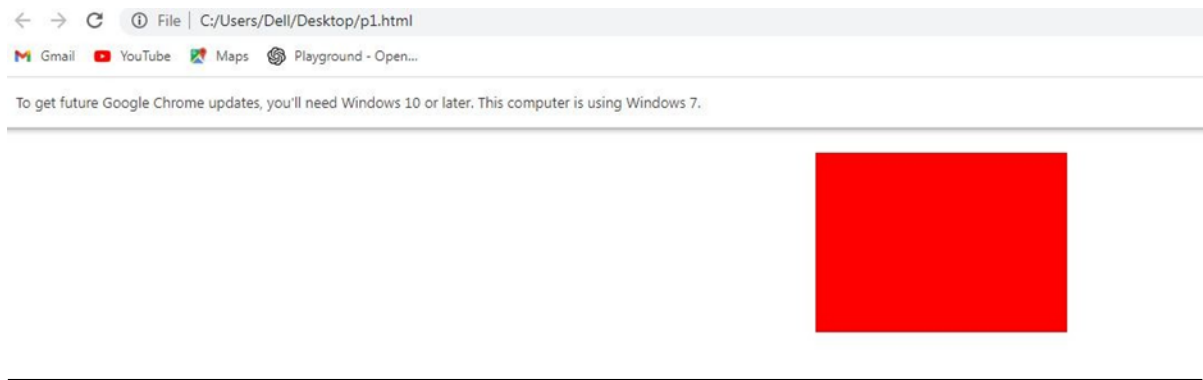
function onMouseOver(){
    var c = document.getElementById("myCanvas");
    var ctx = c.getContext("2d");
    ctx.fillStyle = "red";
    ctx.fillRect(20,20,300,200);
}
</script>
</head>
<body onload="drawRectangle()">
<canvas id="myCanvas" width="300" height="300">

</canvas>

</body>
</html>
```

Output:





7. Write an html program to draw a circle using canvas and to apply the rotations animations on loading the page

```
<!DOCTYPE html>
<html lang="en" xmlns="http://www.w3.org/1999/xhtml">
<head>
  <meta content="text/html; charset=ISO-
    8859-1" http-equiv="content-type">
  <script
    type="application/javascript">
    var centreX = 100; var centreY =
    100; var radius = 75;
    var rotateAngle = 36 * Math.PI /
    180; var startAngle = 0 * Math.PI
    / 180; var endAngle = 36 *
    Math.PI / 180; var counter = 0;
    var animFlag;
    var colours = ["teal", "red", "green", "blue", "yellow", "violet", "orange", "grey", "navy
    blue", "purple"];

    function init() {
      var canvas =
      document.getElementById("canvas"); if
      (canvas.getContext) {
        var ctx = canvas.getContext("2d");
        ctx.lineWidth = 3.0;
        ctx.fillStyle = "orange";
        ctx.fillRect(200, 100, 75,
        20); ctx.fillStyle = "black";
        ctx.font = "15px verdana";
        ctx.fillText("Rotate", 215, 114);
        drawWheel();
      }
    }

    function drawWheel() {
      var canvas =
      document.getElementById("canvas"); if
      (canvas.getContext) {
```

```

var ctx =
canvas.getContext("2d"); for (i
= 0; i < 10; i++) {
    ctx.fillStyle = colours[i];
    ctx.translate(centreX,
    centreY);
    ctx.rotate(rotateAngle);
    ctx.translate(-centreX, -
    centreY); ctx.beginPath();
    ctx.moveTo(centreX,
    centreY);
    ctx.lineTo(centreX + radius, centreY);
    ctx.arc(centreX, centreY, radius, startAngle, endAngle, false);
    ctx.closePath();
    ctx.fill();
}
}
}

```

```

function rotateWheel(rnd) {
    var canvas =
    document.getElementById("canvas"); if
    (canvas.getContext) {
        var ctx =
        canvas.getContext("2d");
        ctx.translate(centreX,
        centreY);
        ctx.rotate(rotateAngle);
        ctx.translate(-centreX, -
        centreY); drawWheel();
        counter++;
        if (counter > rnd)
            { counter = 0;
            clearInterval(anim
            Flag);
            }
        }
    }

function mouseClicked() {
    var rnd = Math.ceil(Math.random() * 100);
    animFlag = setInterval(function () { rotateWheel(rnd) }, 25);
}
window.addEventListener("load", mouseClicked, false);

```

```

</script>
<title>Animation - Moving Banner</title>
</head>
<body onload="init();">
<canvas id="canvas" width="600" height="500"></canvas>

```

```
<br>  
</body>  
</html>
```

Output:

 Gmail  YouTube  Maps  Playground - Open...

To get future Google Chrome updates, you'll need Windows 10 or later. This computer is using Windows 7.



Rotate