11. Program to design a webpage to select and display emoji based on the corresponding mood using canvas.

Main html file:

```
<html>
<head>
<title>Current Mood</title>
</head>
<style>
.mainBoard
{
background-color: rgb(224, 78, 102);
margin: 50px 180px 50px 180px;
height: 600px;
border-radius: 50px;
}
p
{
padding: 50px; font-size:50px;
}
.subBoard
{
background-color: rgb(44, 41, 41);
width: 300px;
height: 300px;
 border-radius: 100px;
}
ul
{
padding: 20px; list-style: none;
}
li{
margin: 30px; font-size: 30px;
}
a{
text-decoration: none; color:white;
}
li:hover
{
background-color: black; border-radius: 30px;
}

</style>
<body>
<div align="center" class="mainBoard">
<p><i>What is your current mood?</i></p>
<div class="subBoard">
<ul>
```

```
<li><a href="happy.html">Happy</a></li>
<li><a href="sad.html">Sad</a></li>
</ul>
</div>
</div>
</body>
</html>
```

happy.html:

```
<html>
<head>
<title>Happy Mood</title>
<style>
.canvas
{
background-color: pink;
margin: 50px 180px 50px 180px; height: 800px;
border-radius: 50px;
}
#HappyCanvas
{
margin:80px; background-color: grey;

border: 2px solid black;
}
p
{
font-size:30px;
}
button
{
padding: 20px; border-radius: 40px;
background-color: grey; border: none;
}
a{ color:black;
font-size: 20px;
text-decoration: none;
}
</style>
<script>
function draw()
{
var canvas = document.getElementById("HappyCanvas");
var ctx = canvas.getContext("2d");
// main circle ctx.beginPath();
```

```
 ctx.lineWidth = 5;
ctx.strokeStyle="black";
ctx.arc(200,200,150,0,2*(Math.PI));
ctx.fillStyle="yellow"; ctx.fill();
ctx.stroke(); ctx.closePath();

//eye1
 ctx.beginPath();
 ctx.lineWidth = 5;
ctx.strokeStyle="black";
ctx.arc(150,170,30,0,2*(Math.PI));
ctx.fillStyle="black";
 ctx.fill();
ctx.stroke();
ctx.closePath();

//eye2
ctx.beginPath();

ctx.lineWidth = 5;
ctx.strokeStyle="black";
ctx.arc(250,170,30,0,2*(Math.PI));
ctx.fillStyle="black"; ctx.fill();
ctx.stroke(); ctx.closePath();

//mouth
 ctx.beginPath();
ctx.lineWidth = 5;
 ctx.strokeStyle="black";
 ctx.moveTo(150,240)
ctx.arc(200,240,50,0,(Math.PI));
ctx.fillStyle="white"; ctx.fill();
ctx.stroke(); ctx.closePath();

//teeth
ctx.beginPath();
 ctx.lineWidth = 6;
ctx.strokeStyle="black";
 ctx.moveTo(155,260)
 ctx.lineTo(245,260)
 ctx.stroke();
 ctx.closePath();
}
</script>
</head>
<body onload="draw()">
<div class="canvas" align="center">
```

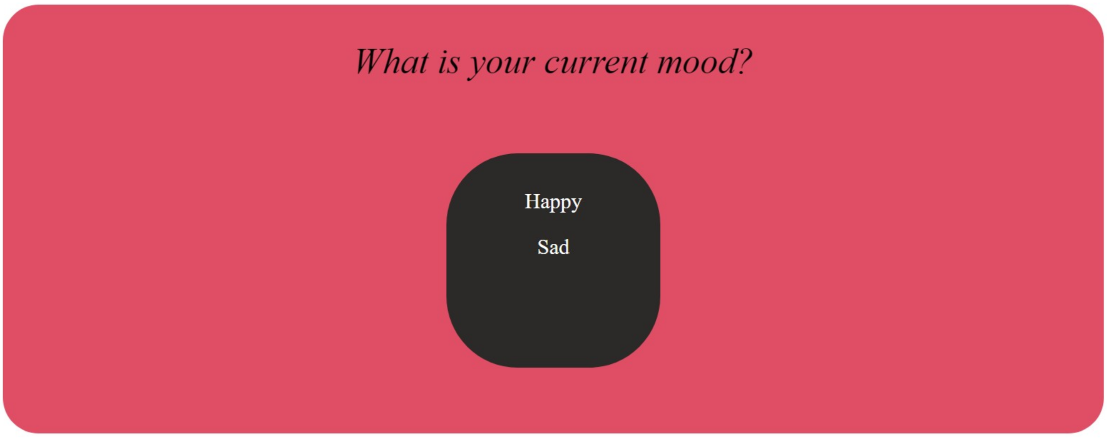
```
<canvas id="HappyCanvas" width="400" height="400"></canvas>
<p class="text">Being happy never goes out of style.</p>
<button><a href="emoji.html"> Back</a></button>
</div>
</body>
</html>
```

sad.html:

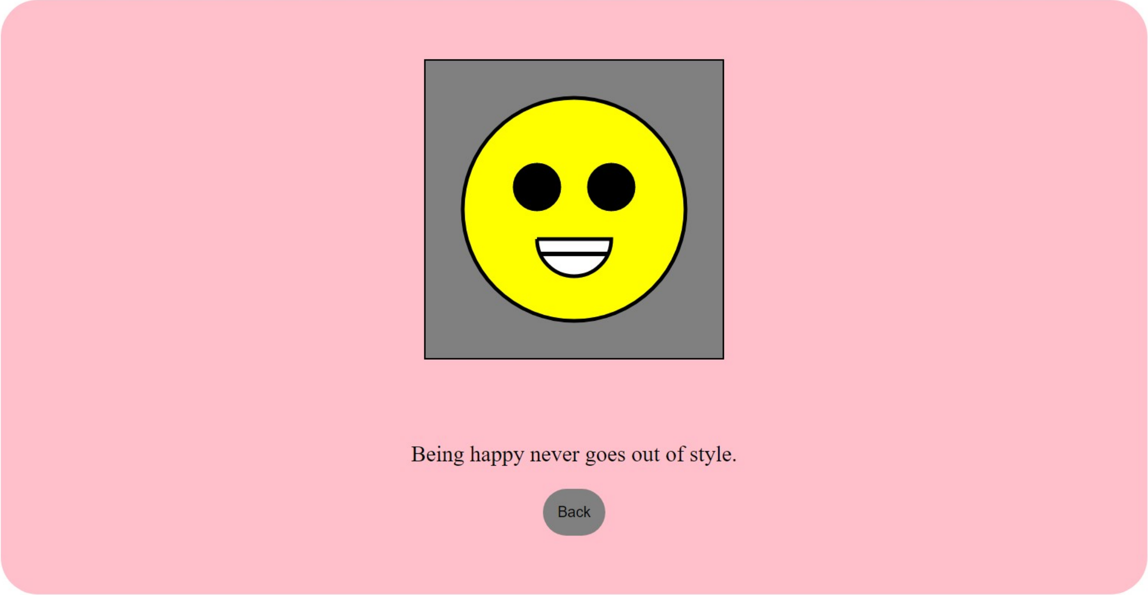
```
<html>
<head>
<title>Sad Mood</title>
<style>
.canvas
{
background-color: rgb(232, 162, 174);
margin: 50px 180px 50px 180px;
height: 800px;
border-radius: 50px;
}
#SadCanvas
{
margin:80px;
background-color: rgb(52, 214, 189);
border: 2px solid rgb(51, 189, 124);
}
p
{
font-size:30px;
}
button
{
padding: 20px;
border-radius: 40px;
background-color: rgb(145, 81, 81);
border: none;
}
a{
color:black;
font-size: 20px;
text-decoration: none;
}
</style>
<script>
function draw()
```

```
{
var canvas = document.getElementById("SadCanvas");
var ctx = canvas.getContext("2d");
// main circle
ctx.beginPath();
ctx.lineWidth = 5;
ctx.strokeStyle="black";
ctx.arc(200,200,150,0,2*(Math.PI));
ctx.fillStyle="yellow";
ctx.fill();
ctx.stroke();
ctx.closePath();


//eye1
ctx.beginPath();
ctx.lineWidth = 5;
ctx.strokeStyle="black";
ctx.arc(150,170,30,0,2*(Math.PI));
ctx.fillStyle="black";
ctx.fill();
ctx.stroke();
ctx.closePath();


//eye2
ctx.beginPath();
ctx.lineWidth = 5;
ctx.strokeStyle="black";
ctx.arc(250,170,30,0,2*(Math.PI));
ctx.fillStyle="black";
ctx.fill();
ctx.stroke();
ctx.closePath();


//mouth
ctx.beginPath();
ctx.lineWidth = 10;
ctx.strokeStyle="black";
ctx.arc(200,280,30,0,(Math.PI),1);
ctx.stroke();
ctx.closePath();


//tear drop
ctx.beginPath();
ctx.lineWidth=5;
ctx.strokeStyle="black"
ctx.moveTo(270,210)
ctx.arc(270,250,20,0,(Math.PI))
ctx.lineTo(270,207)
ctx.fillStyle="lightblue"
ctx.fill();
ctx.stroke();
}
</script>
</head>
<body onload="draw()">
<div class="canvas" align="center">
<canvas id="SadCanvas" width="400" height="400"></canvas>
<p class="text">The clouds of sadness can't stay forever, so take your time and get out of sadness.</p>
<button><a href="emoji.html">  Back</a></button>
</div>
</body>
</html>
```

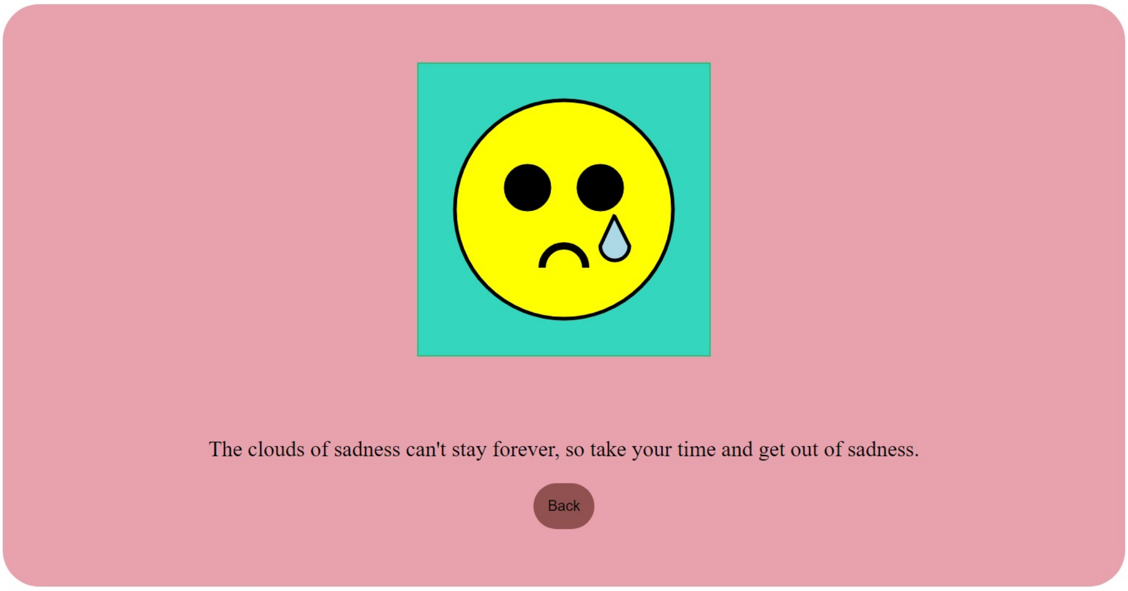
Output:



What is your current mood?



Being happy never goes out of style.



The clouds of sadness can't stay forever, so take your time and get out of sadness.

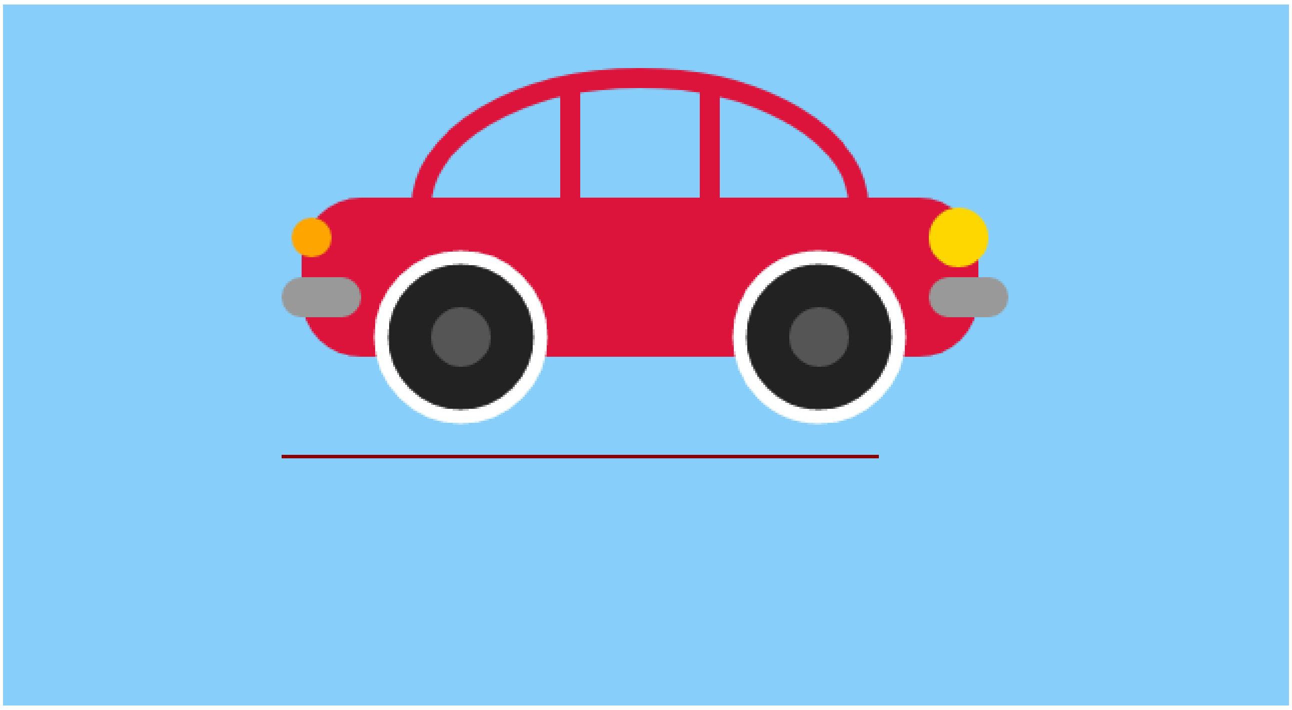
13.Program to create a car using SVG elements.

Program:

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>svg car</title>
  <style type="text/css">
    body {
  padding-top: 30px;
  text-align: center;
  background: lightskyblue;
}

  </style>
</head>
<body>
<svg width="900" height="900">
    <!-- Top -->
      <rect x="70" y="10" width="220" height="130" fill="transparent" rx="150" stroke="crimson"
stroke-width="10" />

    <!-- Body -->
      <rect x="10" y="70" width="340" height="80" fill="crimson" rx="30" />
```

```
   <g>
   <!-- Left line -->
     <line x1="145" y1="10" x2="145" y2="80" stroke="crimson" stroke-width="10"/>

   <!-- Right line -->
     <line x1="215" y1="10" x2="215" y2="80" stroke="crimson" stroke-width="10"/>
   </g>

   <g>
   <!-- Left bumper -->
     <rect x="0" y="110" width="40" height="20" fill="#999" rx="10" />

   <!-- Right bumper -->
     <rect x="325" y="110" width="40" height="20" fill="#999" rx="10" />
   </g>

   <!-- Left wheel -->
   <g>
     <circle r="40px" fill="#222" stroke="white" stroke-width="7" cx="90" cy="140"/>
     <circle r="15px" fill="#555" cx="90" cy="140"/>
   </g>
  <!-- Right wheel -->
   <g>
     <circle r="40px" fill="#222" stroke="white" stroke-width="7" cx="270" cy="140"/>
     <circle r="15px" fill="#555" cx="270" cy="140"/>
      <line x1="0" y1="200" x2="300" y2="200" style="stroke:darkred;stroke-width:2" />
  </g>
 <g>
   <!-- Gold light -->
     <circle r="15px" fill="gold" cx="340" cy="90"/>
         <!-- Orange light -->
     <circle r="10px" fill="orange" cx="15" cy="90"/>
  </g>
</svg>
</body>
</html>
```

Output:

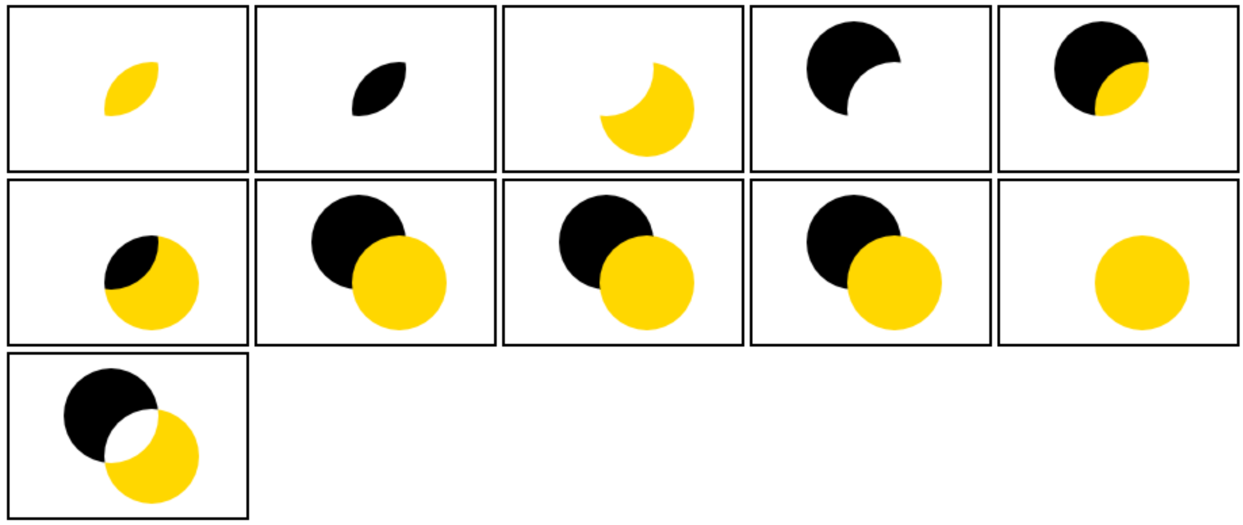


14.Program to crate phases of circle using composition Method.

```
<html>
<body>

   <script>
       ct = ["source-in", "destination-in", "source-out", "destination-out", "source-atop",
          "destination-atop", "detination-out", "lighter", "darker", "copy", "xor"]
       for (i = 0; i < ct.length; i++) {
          document.write("<canvas id=comp" + i + " width=175 height=120 style='border:2px solid 
black;margin:2px;'></canvas>")
          ctx = document.getElementById("comp" + i).getContext("2d")
          ctx.fillStyle = "red"

          ctx.beginPath()
          ctx.arc(75, 45, 35, 0, Math.PI * 2, true)
          ctx.fill()
          ctx.globalCompositeOperation = ct[i]
          console.log(ctx.globalCompositeOperator)
          ctx.fillStyle = "green"
          ctx.beginPath()
          ctx.arc(105, 75, 35, 0, Math.PI * 2, true)
          ctx.fill()
       }
   </script>
</body>
```

```
</html>
```

Output:



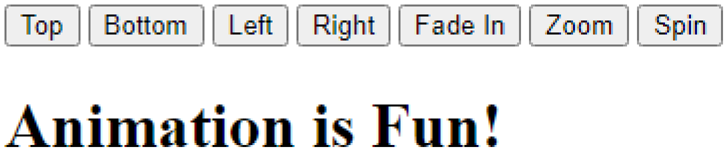
15.Program to create text animation using CSS.

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>Document</title>
    <style>
      #animate-text {
        animation-duration: 1s;
        animation-timing-function: ease-in-out;
        animation-fill-mode: both;
      }

      .animate-top {
        animation-name: slide-top;
      }

      .animate-bottom {
        animation-name: slide-bottom;
      }

      .animate-left {
        animation-name: slide-left;
      }

      .animate-right {
        animation-name: slide-right;
      }

      .animate-fadein {
        animation-name: fade-in;
      }
```

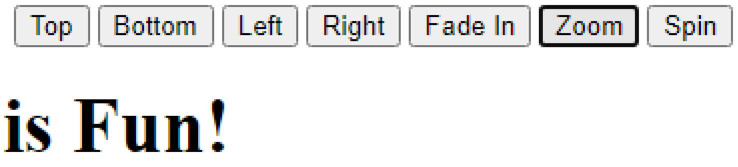
```
.animate-zoom {
  animation-name: zoom;
}

.animate-spin {
  animation-name: spin;
}

@keyframes slide-top {
  from {
    transform: translateY(-50px);
    opacity: 0;
  }
  to {
    transform: translateY(0);
    opacity: 1;
  }
}

@keyframes slide-bottom {
  from {
    transform: translateY(50px);
    opacity: 0;
  }
  to {
    transform: translateY(0);
    opacity: 1;
  }
}

@keyframes slide-left {
  from {
    transform: translateX(-50px);
    opacity: 0;
  }
  to {
    transform: translateX(0);
    opacity: 1;
  }
}

@keyframes slide-right {
  from {
    transform: translateX(50px);
    opacity: 0;
  }
  to {
    transform: translateX(0);
```

```
            opacity: 1;
          }
        }

        @keyframes fade-in {
          from {
            opacity: 0;
          }
          to {
            opacity: 1;
          }
        }

        @keyframes zoom {
          from {
            transform: scale(1);
          }
          to {
            transform: scale(1.2);
          }
        }

        @keyframes spin {
          from {
            transform: rotate(0deg);
          }
          to {
            transform: rotate(360deg);
          }
        }
      </style>
    </head>
    <body>
      <button id="top">Top</button>
      <button id="bottom">Bottom</button>
      <button id="left">Left</button>
      <button id="right">Right</button>
      <button id="fadein">Fade In</button>
      <button id="zoom">Zoom</button>
      <button id="spin">Spin</button>

      <h1 id="animate-text">Animation is Fun!</h1>

      <script>
        const animateText = document.getElementById("animate-text");

        document.getElementById("top").addEventListener("click", () => {
          animateText.classList.remove(
              "animate-bottom",
              "animate-left",
              "animate-right",
              "animate-fadein",
              "animate-zoom",
              "animate-spin"
            );
            animateText.classList.add("animate-top");
          });

          document.getElementById("bottom").addEventListener("click", () => {

            animateText.classList.add("animate-bottom");
          });

          document.getElementById("left").addEventListener("click", () => {

            animateText.classList.add("animate-left");
          });

          document.getElementById("right").addEventListener("click", () => {
            animateText.classList.add("animate-right");
          });

          document.getElementById("fadein").addEventListener("click", () => {
            animateText.classList.add("animate-fadein");
          });

          document.getElementById("zoom").addEventListener("click", () => {
            animateText.classList.add("animate-zoom");
          });

          document.getElementById("spin").addEventListener("click", () => {

            animateText.classList.add("animate-spin");
          });
        </script>
    </body>
</html>
```

Output:



Animation is Fun!



is Fun!