Ex. No. 06	JavaScript Styles and Animation
16.02.2024	

Aim :

The aim of this experiment is to create a web application using JavaScript to manipulate the CSS Styles, Animation and Arrays.

Description:

1. onchange():

In JavaScript, the `onchange()` event handler is instrumental for capturing changes in the values of input elements such as dropdown menus or text fields. This event is commonly employed to trigger functions that respond dynamically to user selections or modifications, enhancing the interactivity of web pages.

2.onclick()

The `onclick()` event handler is essential for executing JavaScript functions when an element is clicked. This functionality is widely utilized in buttons or other clickable elements, allowing developers to define actions that occur in response to user interactions, enhancing the user experience and providing a responsive interface.

3.onkeyup()

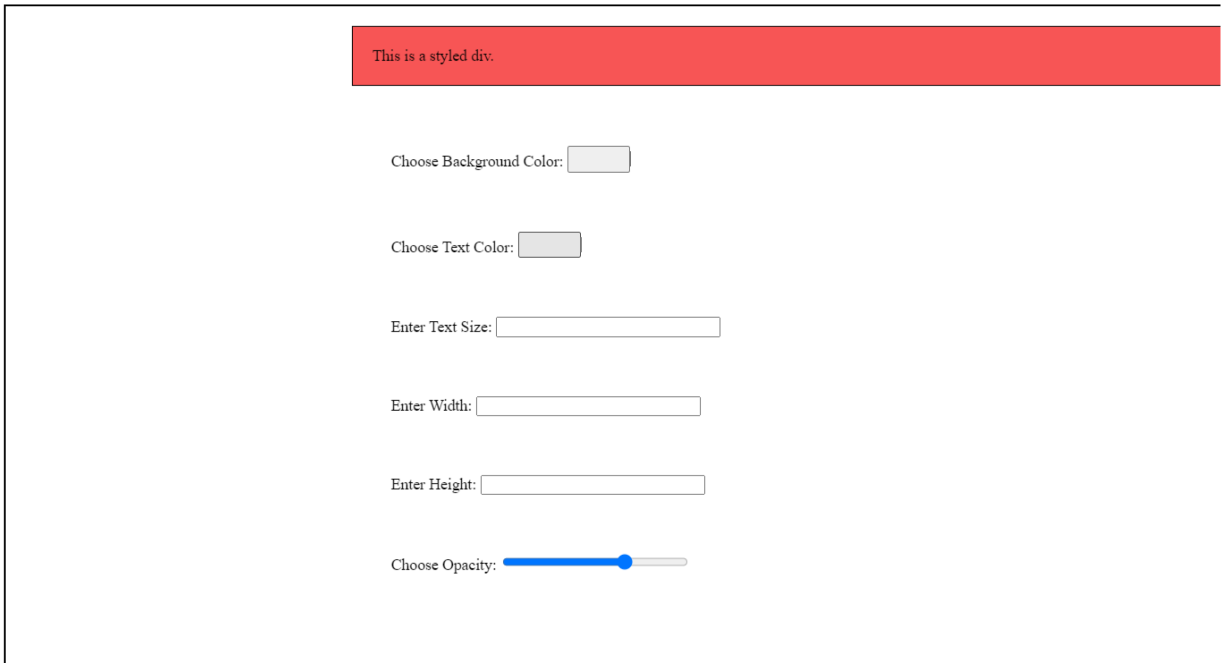
JavaScript's `onkeyup()` event handler is designed to capture the release of a keyboard key after being pressed. This feature is particularly valuable for scenarios where dynamic responses to user input are required, such as implementing live search functionality or updating content as the user types. It facilitates real-time interactions and contributes to a more engaging user interface.

1.HTML code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Style Modification</title>

<style>

    body{

        display: grid;

        justify-content: center;

        align-content: center;

        align-items: center;

        border: 2px solid black

    }

    #styledDiv {

        border: 1px solid black;

        margin: 20px;

        padding: 20px;

    }

    .container{

        width: 60vw;

        height: 80vh;

        border-radius: 10px;

        color: black;

    }

    label, input{

        margin-top: 4%;

        padding-left: 60px;
        }
    </style>
</head
<body>
    <div class="container">
        <div id="styledDiv">This is a styled div.</div>
        <label for="colorPicker">Choose Background Color:</label>
        <input type="color" id="colorPicker" onchange="changeColor()"
        <br><br
        <label for="textColorPicker">Choose Text Color:</label>
        <input type="color" id="textColorPicker" onchange="changeTextColor()">

        <br><br>

        <label for="textSize">Enter Text Size:</label>
        <input type="text" id="textSize" onkeyup="changeTextSize()">

        <br><br>

        <label for="widthInput">Enter Width:</label>
        <input type="text" id="widthInput" onkeyup="changeWidth()">

        <br><br>

        <label for="heightInput">Enter Height:</label>
        <input type="text" id="heightInput" onkeyup="changeHeight()">

        <br><br>

        <label for="opacityRange">Choose Opacity:</label>

        <input type="range" id="opacityRange" min="0" max="1" step="0.1"
onchange="changeOpacity()">

    </div>

    <script>

        function changeColor() {

            var color = document.getElementById("colorPicker").value;

            document.getElementById("styledDiv").style.backgroundColor = color;

        }

        function changeTextColor() {

            var textColor = document.getElementById("textColorPicker").value;

            document.getElementById("styledDiv").style.color = textColor;

        }


        function changeTextSize() {

            var textSize = document.getElementById("textSize").value;

            document.getElementById("styledDiv").style.fontSize = textSize + "px";

        }


        function changeWidth() {

            var widthValue = document.getElementById("widthInput").value;

            document.getElementById("styledDiv").style.width = widthValue + "px";

        }
        function changeHeight() {
            var heightValue = document.getElementById("heightInput").value;
            document.getElementById("styledDiv").style.height = heightValue + "px";
        }

        function changeOpacity() {
            var opacityValue = document.getElementById("opacityRange").value;
            document.getElementById("styledDiv").style.opacity = opacityValue;
        }
    </script
</body>
</html>
```

Output:



This is a styled div.

Choose Background Color:

Choose Text Color:

Enter Text Size:

Enter Width:

Enter Height:

Choose Opacity:

2.HTML Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Moving Car</title>
  <style>
    #carContainer {
      position: relative;
      width: 600px;
      height: 300px;
      margin: 20px auto;
      border: 20px solid #008cff;
      overflow: hidden;
    }
    #car {
      position: absolute;
      top: 50%;
      left: 0;
      transform: translate(0, -50%);
      height:auto;
      width: 200px;
    }
```

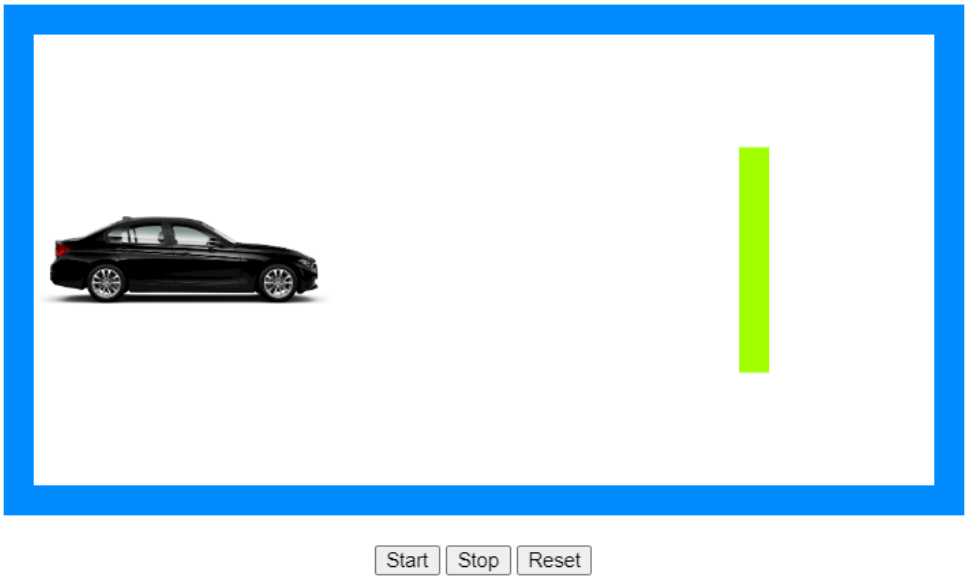
```
#pillar {
    position: absolute;
    top: 50%;
    left: 80%;
    transform: translate(-50%, -50%);
    width: 20px;
    height: 150px;
    background-color: rgb(162, 255, 0);
}

#splashBox {
    display: none;
    position: absolute;
    top: 50%;
    left: 50%;
    transform: translate(-50%, -50%);
    padding: 10px;
    background-color: rgb(0, 255, 238);
    border: 1px solid #000;
}

#buttons {
    text-align: center;
    margin-top: 20px;
}
```

```
  </style>
</head>
<body>
  <div id="carContainer">
    <img id="car" src="pngfind.com-bmw-png-1786020.png" alt="Car Image">
    <div id="pillar"></div>
    <div id="splashBox">Danger</div>
  </div>
  <div id="buttons">
    <button onclick="startCar()">Start</button>
    <button onclick="stopCar()">Stop</button>
    <button onclick="resetCar()">Reset</button>
  </div>
  <script>
    var carInterval;
    var carPosition = 0;

    function startCar() {
      carInterval = setInterval(moveCar, 10);
    }
    function stopCar() {
      clearInterval(carInterval);
      setTimeout(checkCollision, 10);
    }
```

```
function resetCar() {
    clearInterval(carInterval);
    carPosition = 0;
    document.getElementById('car').style.left = carPosition + 'px';
    hideSplashBox();
}

function moveCar() {
    carPosition += 2;
    document.getElementById('car').style.left = carPosition + 'px';
    checkCollision();
}

function checkCollision() {
    var carLeft = carPosition;
    var pillarLeft = document.getElementById('pillar').offsetLeft;

    if (carLeft >= pillarLeft) {
        displaySplashBox();
        clearInterval(carInterval);
    }
}

function displaySplashBox() {
    document.getElementById('splashBox').style.display = 'block';
```

```
        }


        function hideSplashBox() {
            document.getElementById('splashBox').style.display = 'none';
        }
    </script>
</body>
</html>
```

Output:



Result

Thus, the web application successfully achieved its aim of creating an interactive learning environment for participants to explore JavaScript's role in manipulating CSS styles,animations, and arrays. The experiment provided hands-on experience, reinforcing concepts related to web development and JavaScript programming