**Aim:**

Practical – 5

Demonstrate various Ways to handle Events **HTML Source Code:**

<!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>

        body {

            background-color: antiquewhite;

        }

    </style>

</head>

<body>

<h1>Change Message</h1>

<input type="text" id="myInput" onclick="alert('Input was clicked!')" onblur="alert('Input lost focus!')" onmouseover="alert('Mouseover on

Input!')" onkeypress="alert('key pressed!')" >

<script src="./Events.js"></script>

</body>

</html>

**Javascript Source**

const input = document.querySelector("#myInput"); input.addEventListener("click", handleClick); input.addEventListener("blur", handleBlur); input.addEventListener("mouseover", handleMouseOver); input.addEventListener("keypress", handleKeyPress);

function handleClick() { console.log("Input was clicked!");

}

function handleBlur() { console.log("Input lost focus!");

}

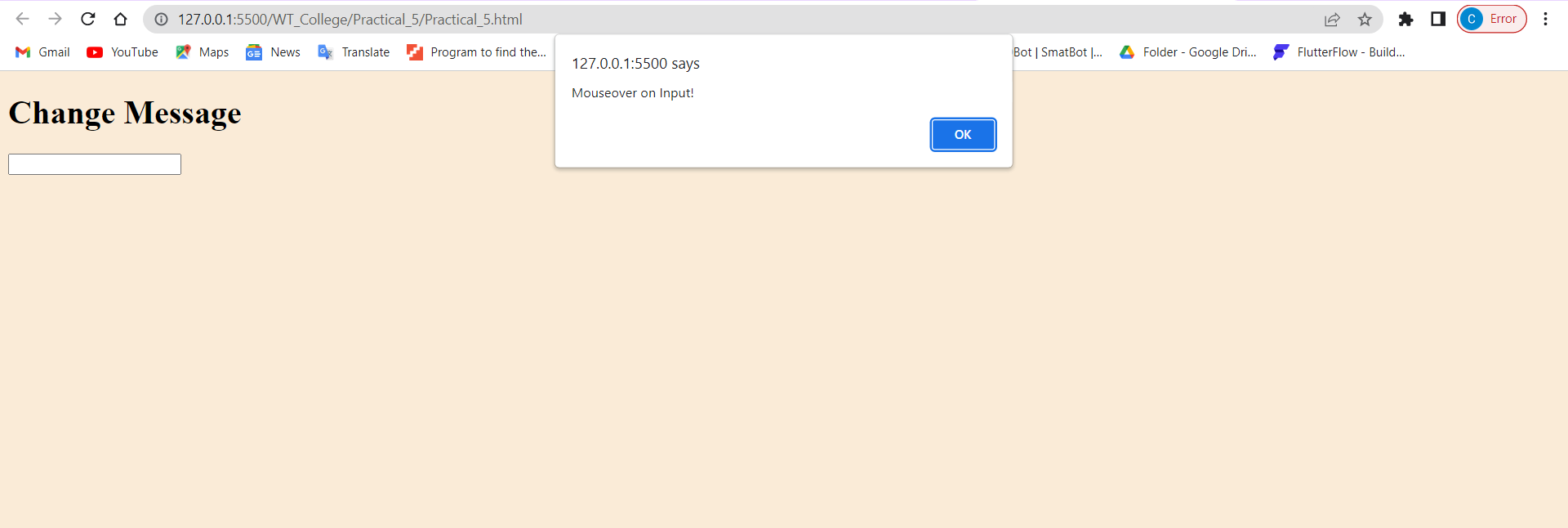
function handleMouseOver() { console.log("Mouseover on Input!");

}

function handleKeyPress(e) { console.log("Key pressed:" + e.key);

}

**Output:**



**Aim:**

**Write the Script to Change the background color randomly after every 1 mins HTML Source Code:**

<!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>Change Background Colour</title>

<script src="./background.js"></script>

</head>

<body>

</body>

</html>

**Javascript Source code:**

let colors = ['red', 'orange', 'yellow', 'green', 'blue', 'indigo', 'violet'];

let currentColorIndex = 0;

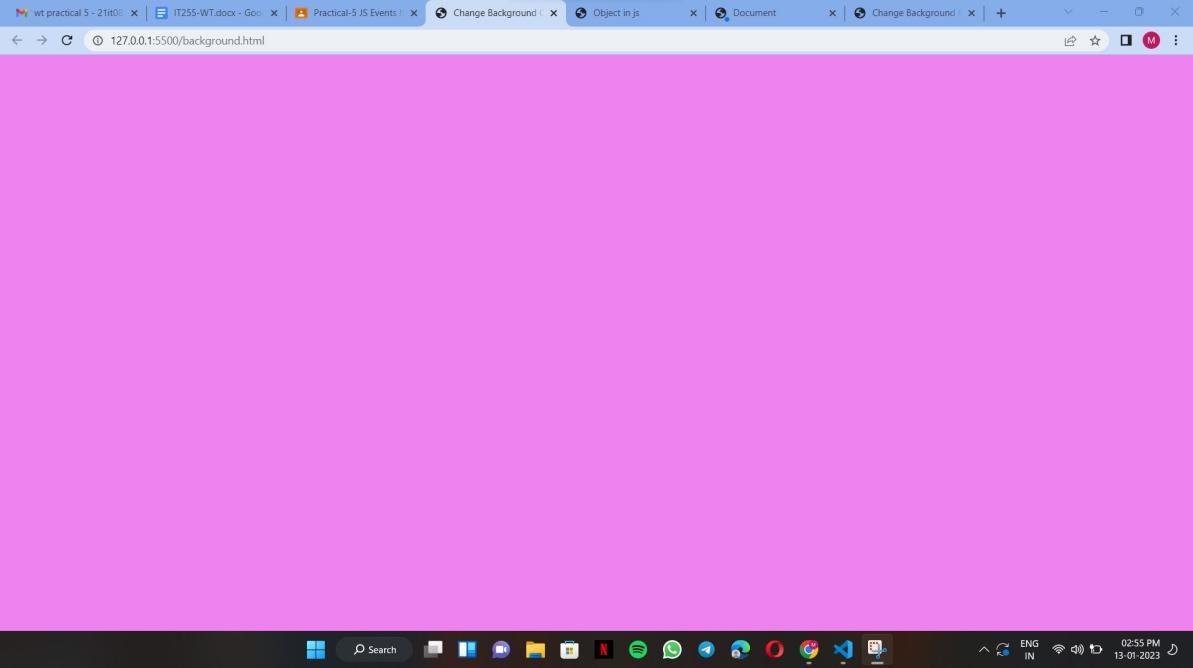
function changeColor() {

let body = document.querySelector('body'); body.style.backgroundColor = colors[currentColorIndex];

currentColorIndex = (currentColorIndex + 1) % colors.length;

}

setInterval(changeColor, 60000);

**Output:**

**AIM:**

<!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>Change Table Colour</title>

<style> table,td{

border: solid black;

}

</style>

</head>

<body>

<h2>Change Table Colour</h2>

<label for="colorInput">Colour Number:</label>

<input type="number" id="colorInput">

<button onclick="changeTableColor()">Change Table Colour</button>

<br><br>

<table id="myTable">

<tr>

<td>ID</td>

<td>Name</td>

<td>Age</td>

</tr>

<tr>

<td>1</td>

<td>abc</td>

<td>19</td>

</tr>

</table>

<script src="./colors in order.js"></script>

</body>

</html>

**Create Arrays of Colors and Apply these colors in order to tr background of table HTML Source Code:**

**Javascript Source code:**

var colors = ["red", "orange", "yellow", "green", "blue", "indigo", "violet"];

var currentColorIndex = 0;

function changeTableColor() {

var table = document.getElementById("myTable");

var colorInput = document.getElementById("colorInput").value; var color = colors[colorInput - 1]; table.style.backgroundColor = color;

}

**Output:**



**AIM:**

**Zoom image on mouseover and zoom out on mouse out HTML Source Code:**

<!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>

</head>

<body>

<img id="myImage"

src="https://images.unsplash.com/photo-1621511101763-32d305e865d4?ixlib=rb

-4.0.3&ixid=MnwxMjA3fDB8MHxzZWFyY2h8MjB8fGhpbGxzfGVufDB8fDB8fA%3D%3D&w=100 0&q=80" onmouseover="zoomIn()" onmouseout="zoomOut()">

<script src="./zoomImage.js"></script>

</body>

</html>

**Javascript Source code:**

function zoomIn() {

var image = document.getElementById('myImage'); image.style.width = "98%";

image.style.height = "98%";

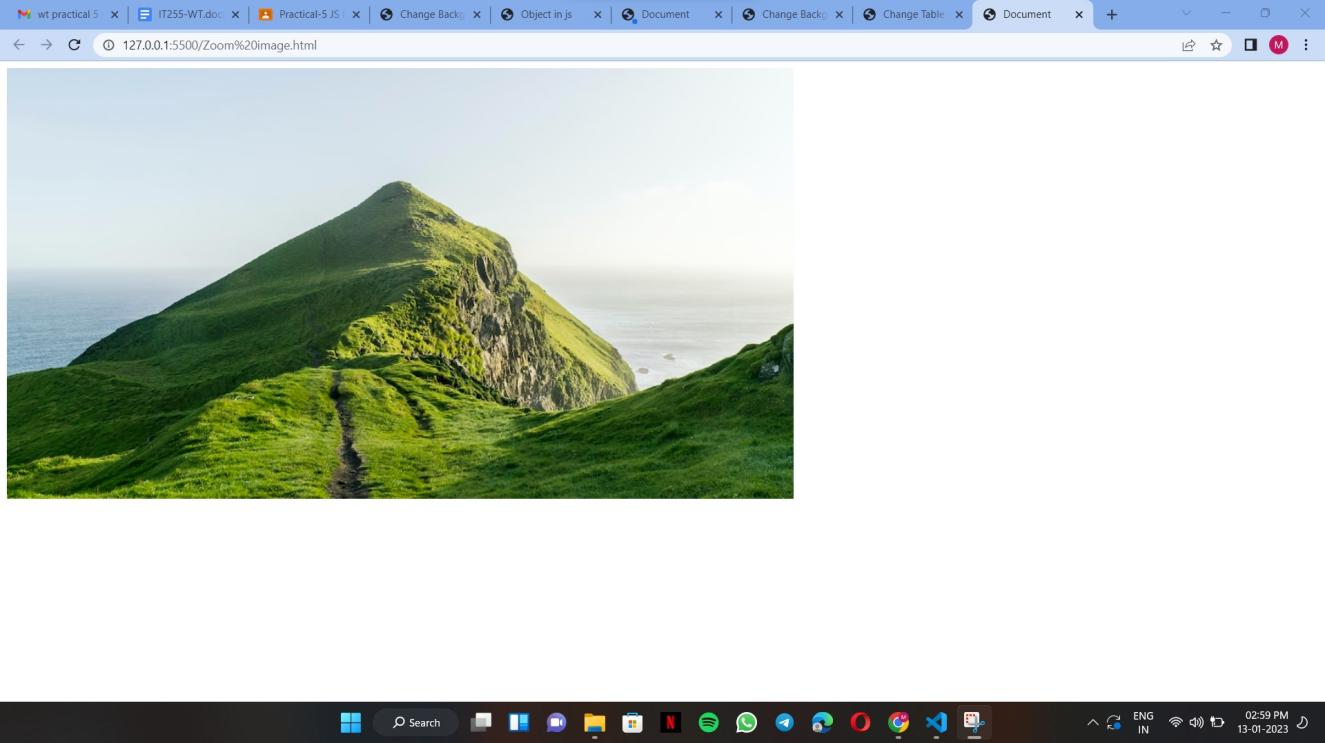
}

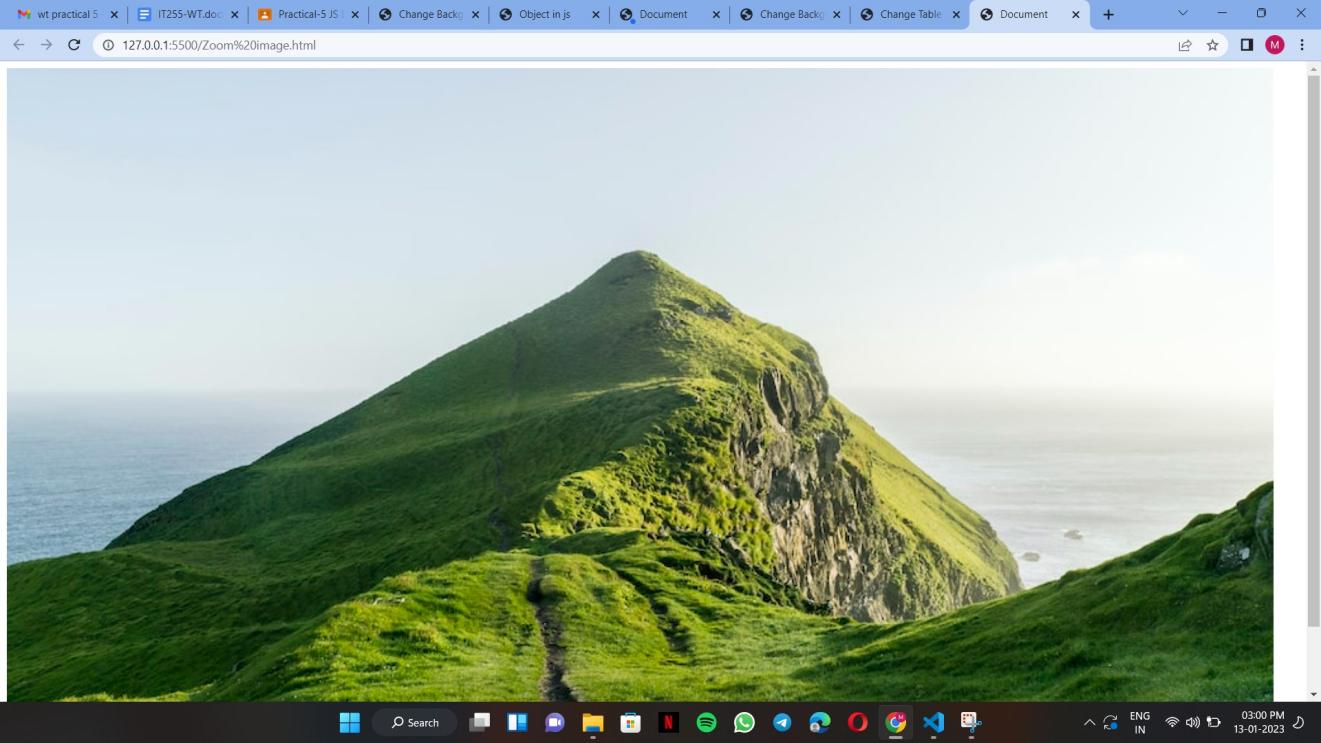
function zoomOut() {

var image = document.getElementById('myImage'); image.style.width = "60%";

image.style.height = "60%";

}

**Output:**



**AIM:**

**Change the location of image based on arrow key of keyboard using callback function HTML Source Code:**

<!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>Move Image</title>

</head>

<body>

<img id="myImage" src="https://cdn.pixabay.com/photo/2015/04/19/08/33/flower-729512 340.jpg " style="position: absolute;">

<script src="./locImg.js"></script>

</body>

</html>

**Javascript Source code:**

let img = document.getElementById("myImage"); document.addEventListener("keydown", moveImage);

function moveImage(event) { switch (event.keyCode) {

case 37:

img.style.left = (img.offsetLeft - 10) + "px"; break;

case 38:

img.style.top = (img.offsetTop - 10) + "px"; break;

case 39:

img.style.left = (img.offsetLeft + 10) + "px"; break;

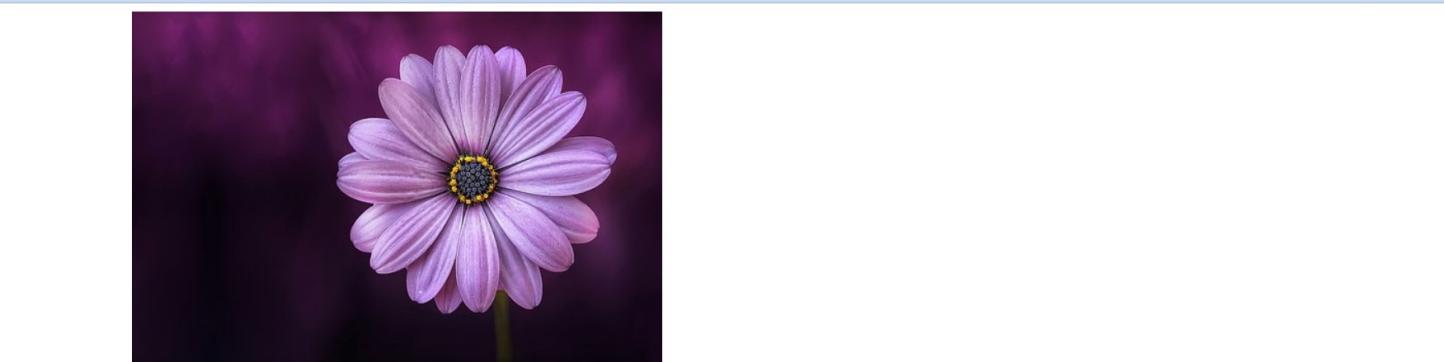
case 40:

img.style.top = (img.offsetTop + 10) + "px"; break;

}

event.preventDefault();

}

**Output:**

**Conclusion:**

A callback function is a function passed into another function as an argument, which is then invoked inside the outer function to complete some kind of routine or action. A timer is used to execute some task after a particular time interval.

Also I have used the timer function to implement the text and it’s working.