**Name: Stephen Chitaranjan B**

**Reg.No: ft38\_542**

### **Subject: JST104 : Synchronous & Asynchronous in**

### **java script**

**Q1: The implementation for your timer application, including HTML, CSS, and JavaScript files.**

1. **index.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Timer Application</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="timer-container">

<h1 id="timer">00:00</h1>

<div class="button-container">

<button id="startBtn">Start</button>

<button id="stopBtn">Stop</button>

<button id="resetBtn">Reset</button>

</div>

</div>

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

</body>

</html>

1. **styles.css**

body {

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

background-color: #f0f0f0;

font-family: Arial, sans-serif;

}

.timer-container {

text-align: center;

background-color: #fff;

padding: 20px;

border-radius: 10px;

box-shadow: 0 4px 10px rgba(0, 0, 0, 0.1);

}

#timer {

font-size: 48px;

margin-bottom: 20px;

}

.button-container button {

padding: 10px 20px;

margin: 5px;

font-size: 16px;

border: none;

border-radius: 5px;

cursor: pointer;

transition: background-color 0.3s;

}

button:hover {

background-color: #e0e0e0;

}

1. **script.js**

let timerInterval;

let secondsElapsed = 0;

const timerDisplay = document.getElementById("timer");

const startBtn = document.getElementById("startBtn");

const stopBtn = document.getElementById("stopBtn");

const resetBtn = document.getElementById("resetBtn");

function updateTimer() {

secondsElapsed++;

const minutes = Math.floor(secondsElapsed / 60);

const seconds = secondsElapsed % 60;

timerDisplay.textContent = `${String(minutes).padStart(2, '0')}:${String(seconds).padStart(2, '0')}`;

}

startBtn.addEventListener("click", () => {

if (!timerInterval) {

timerInterval = setInterval(updateTimer, 1000);

}

});

stopBtn.addEventListener("click", () => {

clearInterval(timerInterval);

timerInterval = null;

});

resetBtn.addEventListener("click", () => {

clearInterval(timerInterval);

timerInterval = null;

secondsElapsed = 0;

timerDisplay.textContent = "00:00";

});

Deployed link: https://github.com/Biju2296/Timer-application.git

**Q2: The implementation for your asynchronous JavaScript slideshow project, including HTML, CSS, and JavaScript files.**

1. **index.html**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Movie Slideshow</title>

<link rel="stylesheet" href="styles.css">

</head>

<body>

<div class="slideshow-container">

<img id="slideshow" src="" alt="Movie Slideshow">

</div>

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

</body>

</html>

1. **styles.css**

body {

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

background-color: #000;

}

.slideshow-container {

width: 80%;

max-width: 600px;

position: relative;

}

#slideshow {

width: 100%;

border-radius: 10px;

box-shadow: 0 4px 20px rgba(0, 0, 0, 0.5);

}

1. **script.js**

const images = [

"https://via.placeholder.com/600x400?text=Movie+1",

"https://via.placeholder.com/600x400?text=Movie+2",

"https://via.placeholder.com/600x400?text=Movie+3",

"https://via.placeholder.com/600x400?text=Movie+4",

"https://via.placeholder.com/600x400?text=Movie+5"

];

let currentIndex = 0;

const slideshow = document.getElementById("slideshow");

// Set the first image as the default image

slideshow.src = images[currentIndex];

// Function to change the image

function changeImage() {

currentIndex = (currentIndex + 1) % images.length; // Loop back to the first image

slideshow.src = images[currentIndex];

}

// Set an interval for the slideshow

setInterval(changeImage, 2000);

Deployed link: