**Лабораторна робота №8**

**Функції setInterval та setTimeout.**

**Мета роботи:** вивчити особливості відкладеного виконання програмного коду.

**Хід роботи:**

***Завдання 1***. Реалізуйте відображення годинника, що відображає поточний час у правому верхньому кутку екрана..

Лістинг програми:

const newClock = document.createElement('div')

document.querySelector('body').appendChild(newClock)

    newClock.classList.add('clock')

const clock=() =>{

    const now = new Date();

    const hours = now.getHours().toString()

    const minutes = now.getMinutes().toString()

    const seconds = now.getSeconds().toString()

    const timeString = `${hours}:${minutes}:${seconds}`

    const clockElement = document.querySelector('.clock')

    clockElement.textContent = timeString;

    setTimeout(clock, 1000)

  }

clock();

Результати виконання:

Рис. 1.1,1.2 – Результат 1 завдання

***Завдання 2***. Реалізуйте таймери зворотного відліку. Розмістіть на сторінці 3 таймери, які мають певне початкове значення часу (години, хвилини та секунди). Кожний таймер повинен містити три кнопки: «Старт», «Стоп», «Скидання». При натисканні на кнопку «Старт», таймер починає зворотній відлік від початкового значення часу до «00:00:00». Досягаючи нульового значення часу таймер повинен зупинитися. При натисканні на кнопку «Стоп» таймер повинен зупинитися. Натискання на кнопку «Скидання» призводить до ініціалізації таймера початковим значенням часу.

Лістинг програми:

const newTimers = document.createElement('div')

document.querySelector('body').appendChild(newTimers)

newTimers.classList.add('timers')

const timers = [];

const timersDiv = document.querySelector('.timers')

function startTimer(id) {

    const timer = timers[id]

    const time = timer.getElementsByTagName('span')[0].innerText.split(':');

    let totalSeconds = parseInt(time[0]) \* 3600 + parseInt(time[1]) \* 60 + parseInt(time[2])

    if (!timer.interval) {

        timer.interval = setInterval(() => {

            totalSeconds--;

            let hours = Math.floor(totalSeconds / 3600)

            let minutes = Math.floor((totalSeconds % 3600) / 60)

            let seconds = totalSeconds % 60

            timer.getElementsByTagName('span')[0].innerText =

                (hours < 10 ? '0' + hours : hours) + ':' +

                (minutes < 10 ? '0' + minutes : minutes) + ':' +

                (seconds < 10 ? '0' + seconds : seconds);

            if (totalSeconds <= 0) {

                stopTimer(id);

            }

        }, 1000);

    }

}

function stopTimer(id) {

    const timer = timers[id];

    clearInterval(timer.interval);

    timer.interval = null;

}

function resetTimer(id) {

    const timer = timers[id];

    const time = timer.getElementsByTagName('span')[0].innerText.split(':');

    stopTimer(id);

    for (let i = 0; i < 3; i++) {

        const timer = document.querySelector('.timer');

        timer.className = 'timer';

        timer.innerHTML = `

            <span>${i + 1}:00:00</span>

            <button onclick="startTimer(${i})">Start</button>

            <button onclick="stopTimer(${i})">Stop</button>

            <button onclick="resetTimer(${i})">Reset</button>

        `;

        timersDiv.appendChild(timer);

        timers.push(timer);

    }

}

for (let i = 0; i < 3; i++) {

    const timer = document.createElement('div');

    timer.className = 'timer';

    timer.innerHTML = `

        <span>${i + 1}:00:00</span>

        <button onclick="startTimer(${i})">Start</button>

        <button onclick="stopTimer(${i})">Stop</button>

        <button onclick="resetTimer(${i})">Reset</button>

    `;

    timersDiv.appendChild(timer);

    timers.push(timer);

}

Результати виконання:

Рис. 2.1,2.2 – Результат виконання 2 завдання

***Завдання 3.*** Створіть просте слайд-шоу графічних зображень, яке буде автоматично змінювати картинки через певний інтервал часу.

Лістинг програми:

let slideIndex = 0

const slideshow=()=> {

    let slides = document.querySelectorAll('.slide')

    for (let i = 0; i < slides.length; i++)

        slides[i].style.display = "none"

    slideIndex++

    if (slideIndex > slides.length)

        slideIndex = 1

    slides[slideIndex-1].style.display = "block"

    setTimeout(slideshow, 2000)

}

slideshow()

Результат виконання

Рис 3.1,3.2 – Результат виконання завдання

***Завдання 4.*** Реалізуйте автоматичне переміщення декількох блоків по екрану. Блоки повинні починати рухатися у випадкових напрямках. Досягаючи границь вікна браузера блоки повинні починати рух у зворотному напрямку.

Лістинг програми:

const numBlocks = 5;

const speed = 2;

let blocks = [];

function createBlocks() {

    for (let i = 0; i < numBlocks; i++) {

        const block = document.createElement('div')

        block.classList.add('block')

        document.body.appendChild(block)

        blocks.push(block)

    }

}

function getRandomInt(min, max) {

    return Math.floor(Math.random() \* (max - min + 1)) + min;

}

function initBlocks() {

    for (let block of blocks) {

        block.style.left = getRandomInt(0, window.innerWidth - block.clientWidth) + 'px'

        block.style.top = getRandomInt(0, window.innerHeight - block.clientHeight) + 'px'

        block.velocityX = getRandomInt(-speed, speed)

        block.velocityY = getRandomInt(-speed, speed)

    }

}

function updateBlocks() {

    for (let block of blocks) {

        let x = parseInt(block.style.left) + block.velocityX

        let y = parseInt(block.style.top) + block.velocityY

        if (x < 0 || x > window.innerWidth - block.clientWidth) {

            x = x < 0 ? 0 : window.innerWidth - block.clientWidth;

            block.velocityX = -block.velocityX;

        }

        if (y < 0 || y > window.innerHeight - block.clientHeight) {

            y = y < 0 ? 0 : window.innerHeight - block.clientHeight;

            block.velocityY = -block.velocityY;

        }

        block.style.left = x + 'px';

        block.style.top = y + 'px';

    }

}

const startAnimation=() =>{

    createBlocks();

    initBlocks();

    setInterval(updateBlocks, 20)

}

startAnimation()

Результати виконання:

Рис. 4 – Результат виконання завдання

***Висновки.*** вивчив особливості відкладеного виконання програмного коду