Javascript script

```
// Function to update the current day of the week and time
const updateDateTime = () => {
    const now = new Date();

    const daysOfWeek = [
        "Sunday",
        "Monday",
        "Tuesday",
        "Wednesday",
        "Thursday",
        "Friday",
        "Saturday",
    ];
    const dayOfWeek = daysOfWeek[now.getDay()];
    document.getElementById("dayOfWeek").textContent = dayOfWeek;

    const time = now.toLocaleTimeString("en-US", {
        day: "numeric",
        month: "long",
        year: "numeric",
        hour: "numeric",
        minute: "numeric",
        second: "numeric",
        hour12: true,
    });
    document.getElementById("time").textContent = time;
};

// Update the time immediately on page load
updateDateTime();

// Update the time every second (1000 milliseconds)
setInterval(updateDateTime, 1000);

// Function to toggle the theme
const toggleTheme = () => {
    const themeSwitch = document.getElementById("themeSwitch");
    const isChecked = themeSwitch ? themeSwitch.checked : false;

    if (isChecked) {
        setTheme("dark");
    } else {
        setTheme("light");
    }
};

// Function to change the background color based on the selected theme
const changeBackgroundColor = () => {
    const theme = localStorage.getItem("theme");
    document.documentElement.classList.toggle("dark", theme === "dark");

    const contentContainer = document.getElementById("contentContainer");
    contentContainer.classList.toggle("dark", theme === "dark");

    const links = document.getElementById("links").getElementsByTagName("a");
    for (let i = 0; i < links.length; i++) {
        links[i].classList.toggle("dark", theme === "dark");
    }
};

// Function to set the theme in local storage
const setTheme = (themeName) => {
    localStorage.setItem("theme", themeName);
    changeBackgroundColor();
};

// Retrieve the current theme from local storage, default to "light" if not set
const currentTheme = localStorage.getItem("theme") || "light";

// Set the initial theme and update the toggle switch accordingly
setTheme(currentTheme);
document.getElementById("themeSwitch").checked = currentTheme === "dark";

// Function to initialize visit count and theme on page load
const initialize = () => {
    if (localStorage) {
        const count = localStorage.getItem("visitCount") || 0;
        const newCount = parseInt(count) + 1;
        localStorage.setItem("visitCount", newCount);
        updateDOM();
    }
};
```

```
// Function to update visit count and theme on the page
const updateDOM = () => {
    const count = localStorage.getItem("visitCount") || 0;
    const theme = localStorage.getItem("theme");
    document.getElementById("visitCount").textContent = `Visit count ${count}`;
    document.getElementById("themeSwitch").checked = theme === "dark";
};

// Function to set the theme when a color preference is selected
const setColorPreference = (color) => {
    setTheme(color);
};

// Attach the initialize function to the window's onload event
window.onload = initialize;

// Function to open the modal
function openModal() {
    const modal = document.getElementById("modal");
    modal.style.display = "block";
}

// Function to close the modal
function closeModal() {
    const modal = document.getElementById("modal");
    modal.style.display = "none";
}

// Open file dialog when the dropzone area is clicked
const dropzone = document.getElementById("dropzone");
dropzone.addEventListener("click", () => {
    const fileInput = document.createElement("input");
    fileInput.type = "file";
    fileInput.id = "fileInput";
    fileInput.accept = "image/*";
    fileInput.addEventListener("change", (event) => {
        handleFileUpload(event.target.files);
    });
    fileInput.click();
});

// Function to preview the uploaded image file
function previewImage(file) {
    const preview = document.getElementById("previewImage");
    const reader = new FileReader();
    reader.readAsDataURL(file);
    reader.onloadend = () => {
        preview.src = reader.result;
    };
}

// Function to handle the file upload
function handleFileUpload(files) {
    const file = files[0];
    if (file) {
        previewImage(file);
    }
}

// Function to handle the form submission
function handleSubmit(event) {
    event.preventDefault();
    const name = document.getElementById("name").value;
    const regNo = document.getElementById("email").value;
    const image = document.getElementById("previewImage").src;

    // Create an object with the form data
    const user = {
        name: name,
        regNo: regNo,
        image: image,
    };

    // Save the user data to local storage
    const users = JSON.parse(localStorage.getItem("users")) || [];
    users.push(user);
    localStorage.setItem("users", JSON.stringify(users));
    console.log(users);

    // Close the modal
    closeModal();
}

// Add an event listener to handle form submission
const submitButton = document.getElementById("submit");
submitButton.addEventListener("click", handleSubmit);
```

```
// Add an event listener to close the modal when the close button is clicked
const closeButton = document.querySelector(".close");
closeButton.addEventListener("click", closeModal);

// Function to display the users in a table
const displayUsers = () => {
    const users = JSON.parse(localStorage.getItem("users")) || [];
    const tbody = document.getElementById("tbody");
    tbody.innerHTML = "";
    users.forEach((user) => {
        const tr = document.createElement("tr");
        const td1 = document.createElement("td");
        const td2 = document.createElement("td");
        const td3 = document.createElement("td");
        const img = document.createElement("img");
        img.src = user.image;
        img.alt = user.name;
        img.width = 100;
        img.height = 100;
        td1.textContent = user.name;
        td2.textContent = user.regNo;
        td3.appendChild(img);
        tr.appendChild(td1);
        tr.appendChild(td2);
        tr.appendChild(td3);
        tbody.appendChild(tr);
    });
};

// Add an event listener to the window to display the users when the page loads
window.addEventListener("load", displayUsers);

// clear storage
document.getElementById("ClearStorage").addEventListener("click", function () {
    localStorage.clear();
    window.location.reload();
});
```