Personal Album Website

Project Title: Personal Album Website

Technologies Used: HTML, CSS, JavaScript

Author: Manish Kumar (12201560)

Submitted To: Sandeep Kaur

---

1. Introduction

The Personal Album Website is a web-based application designed to display a collection of images in an organized and visually appealing manner. This project was developed using HTML, CSS, and JavaScript to create an interactive and responsive photo gallery.

Objectives:

- Create a user-friendly photo gallery
- Implement smooth navigation between images
- Ensure the website is responsive across different devices
- Provide an aesthetically pleasing design

2. Features

1. **Responsive Design**: Works on desktops, tablets, and mobile devices
2. **Image Gallery**: Displays images in a grid layout
3. **Lightbox Feature**: Clicking on an image opens a larger view with navigation controls
4. **Search Functionality**: Allows users to search for specific images
5. **Filtering Options**: Users can filter images by categories (e.g., Nature, Travel, Portraits)

3. Technologies Used

Technology	Purpose
HTML	Structure of the webpage
CSS	Styling and layout
JavaScript	Interactive features (lightbox, filtering, search)

---

4. Implementation Details

4.1 HTML Structure

The website consists of:

- A header with a title and search bar
- A filter section with category buttons
- A main gallery section displaying images in a grid
- A lightbox modal for enlarged image viewing

4.2 CSS Styling

- **Flexbox & Grid Layout** for responsive design
- **Hover Effects** on images for better user interaction
- **Modal Styling** for the lightbox feature

4.3 JavaScript Functionality

- **Image Filtering:** Filters images based on selected categories
- **Search Function:** Searches images by keywords
- **Lightbox Controls:** Allows navigation between images in full-screen mode

---

5. Code Snippets

5.1 HTML (index.html)

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Personal Album</title>
    <link rel="stylesheet" href="styles.css">
</head>
<body>
    <header>
        <h1>My Photo Album</h1>
        <input type="text" id="search" placeholder="Search images...">
    </header>
    <div class="filters">
        <button class="filter-btn active" data-filter="all">All</button>
        <button class="filter-btn" data-filter="nature">Nature</button>
        <button class="filter-btn" data-filter="travel">Travel</button>
    </div>
    <div class="gallery" id="gallery"></div>
    <div class="lightbox" id="lightbox">
        <span class="close">&times;</span>
        <img class="lightbox-img" id="lightbox-img">
    </div>
    <script src="script.js"></script>
</body>
</html>
```

5.2 CSS (styles.css)

```css
body {
    font-family: Arial, sans-serif;
    margin: 0;
    padding: 20px;
    background: #f4f4f4;
}
.gallery {
    display: grid;
    grid-template-columns: repeat(auto-fill, minmax(200px, 1fr));
    gap: 15px;
}
.gallery img {
    width: 100%;
    border-radius: 8px;
    cursor: pointer;
    transition: transform 0.3s;
}
.gallery img:hover {
    transform: scale(1.05);
}
.lightbox {
    display: none;
    position: fixed;
    top: 0;
    left: 0;
    width: 100%;
    height: 100%;
    background: rgba(0, 0, 0, 0.8);
    z-index: 1000;
}
.lightbox-img {
    max-width: 80%;
    max-height: 80%;
    margin: 5% auto;
    display: block;
}
```

5.3 JavaScript (script.js)

javascript

```
document.addEventListener('DOMContentLoaded', () => {
    const gallery = document.getElementById('gallery');
    const lightbox = document.getElementById('lightbox');
    const lightboxImg = document.getElementById('lightbox-img');
    const closeBtn = document.querySelector('.close');
    const filterBtns = document.querySelectorAll('.filter-btn');
    const searchInput = document.getElementById('search');

    // Sample images data
    const images = [
        { src: 'img1.jpg', category: 'nature' },
        { src: 'img2.jpg', category: 'travel' },
        // Add more images
    ];

    // Display images in gallery
    function displayImages(filter = 'all', searchTerm = '') {
        gallery.innerHTML = '';
        const filteredImages = images.filter(img =>
            (filter === 'all' || img.category === filter) &&
            (searchTerm === '' || img.src.toLowerCase().includes(searchTerm))
        );
        filteredImages.forEach(img => {
            const imgElement = document.createElement('img');
            imgElement.src = img.src;
            imgElement.addEventListener('click', () => openLightbox(img.src));
            gallery.appendChild(imgElement);
        });
    }

    // Open lightbox
    function openLightbox(src) {
        lightbox.style.display = 'block';
        lightboxImg.src = src;
    }

    // Close lightbox
    closeBtn.addEventListener('click', () => {
        lightbox.style.display = 'none';
    });

    // Filter buttons event
    filterBtns.forEach(btn => {
        btn.addEventListener('click', () => {
            filterBtns.forEach(b => b.classList.remove('active'));
            btn.classList.add('active');
```

```
            displayImages(btn.dataset.filter, searchInput.value.toLowerCase());
        });
    });

    // Search functionality
    searchInput.addEventListener('input', () => {
        const activeFilter = document.querySelector('.filter-btn.active').dataset.filter;
        displayImages(activeFilter, searchInput.value.toLowerCase());
    });

    // Initial display
    displayImages();
});
```

---

6. Challenges Faced

- Making the gallery **fully responsive**
- Implementing **smooth transitions** in the lightbox
- Ensuring **cross-browser compatibility**

7. Future Enhancements

- Adding **user authentication** for private albums
- Implementing **image upload functionality**
- Adding **slideshow mode** for automatic image display

---

8. Conclusion

The **Personal Album Website** project successfully demonstrates the use of **HTML, CSS, and JavaScript** to create an interactive photo gallery. It provides a seamless user experience with filtering, searching, and a lightbox feature.

Submitted by:

Manish Kumar (12201560)

Section: K22CK

Submitted to: Sandeep Kaur