<https://github.com/Mukhammedali223/Travel-web.git>

**Assignment 6 – Advanced JavaScript and DOM Manipulation**  
**Members:** Mukhammedali Khassenov, Shakarim Ainatayev, Kuanysh Seitzhan  
**Group:** SE-2414

**Objective**

The purpose of this assignment is to demonstrate advanced JavaScript techniques such as DOM manipulation, event handling, animations, sound integration, and responsive design using Bootstrap. The project consists of several pages including Home, Places, Gallery, Tips, and Contact, all styled and scripted to create an interactive user experience.

**1. DOM Manipulation**

Dynamic generation of content was implemented to create elements directly through JavaScript. The “Places” page loads cards dynamically from an array of objects. Each card is created and inserted into the DOM using createElement and appendChild

places.forEach(place => {

const card = document.createElement("div");

card.className = "col-md-4 mb-4";

card.innerHTML = `

<div class="card h-100">

<img src="${place.image}" class="card-img-top">

<div class="card-body">

<h5>${place.name}</h5>

<p>${place.description}</p>

</div>

</div>`;

placesContainer.appendChild(card);

});

This proves that the elements are not static in HTML, but generated dynamically through JavaScript

**2. Event Handling**

Several types of user interactions were implemented. A theme toggle button changes between light and dark mode by modifying CSS classes. Other event handlers include form submission and gallery navigation

document.getElementById("themeToggle").addEventListener("click", () => {

document.body.classList.toggle("dark-mode"); });

This ensures that the page responds immediately to user input without reloading

**3. Advanced JavaScript Features**

Arrays, objects, functions, and localStorage are used to handle data and store user preferences such as selected theme or ratings. Callback functions and loops are used to process and display data dynamically

localStorage.setItem("theme", currentTheme);

This implementation demonstrates the ability to handle persistent data and enhance user experience through memory of previous actions

**4. Sound Integration**

Sound effects are included for specific user actions such as clicking buttons or submitting forms. The Web Audio API and the Audio object are used to play sounds dynamically

function playClickSound() {

const audio = new Audio("sounds/click.mp3");

audio.play();

}

This feature adds interactivity and improves user engagement

**5. Animation**

CSS transitions and transformations are used to create smooth animations for UI elements such as cards and gallery images. Hovering over an element triggers animation to enhance the visual appeal

.card:hover {

transform: scale(1.03);

transition: 0.3s ease;

}

Animations make the website visually dynamic and modern.

**6. Responsive Design**

Bootstrap 5 is used to ensure full responsiveness of the layout across different devices and screen sizes. Containers, rows, and columns are implemented according to Bootstrap’s grid system.

<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">

This guarantees a consistent layout on desktops, tablets, and mobile devices.

**7. Page Structure and File Organization**

All HTML, CSS, and JavaScript are separated into different files for better organization and maintainability  
Files included:

index.html

places.html

gallery.html

tips.html

contact.html

css/style.css

js/script.js

Each file serves a specific function, with clear structure and comments in code for easier understanding.

**Conclusion**

All assignment criteria have been successfully completed.  
The project demonstrates:

Complete DOM manipulation

Proper event handling

Usage of advanced JavaScript features

Integration of sound and animation

Responsive design with Bootstrap

Clean structure with separated files