CSI 3140 - Summer 2025

LAB 5: Interactive Blog

Due Date: Monday Jun 30th at 11:59PM EST

Compress all your files and upload the Zip file to the Brightspace. Please name the file using the following format: `CSI3140_Lab_<lab_number>_<student_id_1>_<student_id_2>.zip`.

You will build an "Interactive Blog" page using your knowledge of HTML, CSS, and JavaScript. The primary focus of this lab is to apply your understanding of JavaScript **Arrays**, **Objects**, the **DOM**, and **Event Handling** to create a dynamic, single-page application.

Instructions:

Create a single web page that displays a list of blog posts and allows users to add new posts through a form. The entire application will run client-side.

1. User Interface (HTML & CSS):

Create a user interface. The interface must contain the following elements:

- A `div` element to serve as a container for all the blog posts (e.g., `<div id="postsContainer"></div>`).
- A `<form>` element for users to submit new posts.
- Two `input` elements to get the post title and the author.
- A `<textarea>` element for the post content.
- A `<button>` element to submit the form and create the post.

2. Program Logic (JavaScript):

- You must use a `global` variable to store the blog posts. This should be an **Array of Objects**. Each object in the array represents a single post and should have properties for `title`, `author`, `date`, and `content`.
- Create **displayPosts() function**:
  - This function should not take any arguments.
  - It must clear the `postsContainer` div to prevent duplicate content.
  - It should `loop` through the global array of blog posts.

- o For each post `object`, it must dynamically create the necessary HTML elements (`<h3>`, `<p>`, etc.) and append them to the `postsContainer`.
- o A **click event listener** must be added to each post's title. When clicked, this listener will trigger an `alert()` showing the full content of that post.
- Create **handleAddPost(event) function:** This function should be triggered when the form's submit button is clicked.
  - o It must prevent the form's default page-reload behavior.
  - o It should read the values from the `title`, `author`, and `content` input fields.
  - o It should create a new post object (including the current `new Date()`) and push it to the `global` blog posts `array`.
  - o After adding the new post, it should call `displayPosts()` to refresh the content on the `page` and then clear the `form` fields.

3. Application Flow:

- When the page first loads, the initial set of blog posts stored in your JavaScript `array` should be automatically displayed.
- The user can `add` a new post at any time using the form.
- The `list` of posts on the page should be updated after a new post is submitted.
- The user can `click` on the title of any post to view its full content in an `alert` dialog.

Example:

My Simple Blog

A client-side blog powered by JavaScript

Latest Posts

Understanding the DOM

By Hesam Nasiri on June 15, 2025

The Document Object Model (DOM) is a programming interface for web documents. It represents the page...

JavaScript Event Handling

By John Smith on June 16, 2025

Event handling is the cornerstone of interactive web pages. It involves listening for actions—like c...

Create a New Post

Title

Author

Content

Create Post

This page says

Event handling is the cornerstone of interactive web pages. It involves listening for actions—like clicks, mouse movements, or key presses—and responding to them with a function. We can attach event listeners to any DOM element to make our pages dynamic and responsive to user input.

OK

Latest Posts

Understanding th

By Hesam Nasiri on June 15, 2025

The Document Object Model (DOM) is a programming interface for web documents. It represents the page...

JavaScript Event Handling

By John Smith on June 16, 2025

Event handling is the cornerstone of interactive web pages. It involves listening for actions—like c...

ate a New Post

Author

Content

Create Post