___

Submission Requirements:

1. Create an Express application.
2. When you are finished:
   a. Delete the node_modules/ folder
   b. Rename the project to **A2_FIRSTNAME**. Replace **FIRSTNAME** with your name, example: **A2_PAOLO**
   c. Create a zip file containing the entire project folder:
      i. Windows: Right click on the project folder > Send To > Compressed (zip) file
      ii. Mac: Right click on the project folder > Compress
3. Submit the zip file to the course website.

Implementation Guidelines:

Your overall implementation must reflect the coding practices demonstrated in class. Key points include:

- When declaring variables, use let and const

- When declaring functions, use the const arrow function syntax described in class

- You **may** use the array shortcut functions: map(), reduce(), find(), forEach(), and filter()

Academic Integrity:

This is an individual assessment.   While it is normal to want to study in a group, there is a fine line between helpful group discussion and academic dishonesty. It's fine to talk through general strategies or high-level approaches, but you must write your own code and avoid sharing or copying implementation details. If studying with others leads you to submit solutions that look nearly identical, then very likely you've crossed the boundary of what is acceptable group discussion.

When creating your assignment, please note the following:

What is allowed:

- Using course materials or internet resources to refresh your memory on basic Javascript syntax or concepts

Not permitted:

- Reposting any part of the assessment to online forums or homework help websites
- Contract plagiarism: Purchasing a solution, or completing a solution for compensation
- Sharing or receiving source code, references, or assistance from others

Usage of Artificial Intelligence (AI) tools:

- Using an AI to generate code on your behalf is NOT permitted.
- Asking an AI to solve a "bug" or "error" for you is NOT permitted. There is a fine line between asking an AI to explain to you the **meaning** of an error, and asking it for a solution to your error.
- The assignment must reflect your own understanding of the course material. Your submission should be based on what you have learned and can apply, not on what an AI system produces

You have been hired to create a web application that represents a travel journal about memorable locations in a destination of your choice. The app must provide these features

The app must provide these features:
- Show all memorable locations
- Delete a memorable location

Using the techniques demonstrated in class, implement this application using Express.

SPECIFICALLY:

1. Choose a Travel Destination

Select a destination of your choice. Users will add memorable locations that are located in that destination.

2. Implement data source

Create a Javascript module that contains a *list* of locations in your chosen travel destination.

Each location must have:

- Id: an **integer** that uniquely identifies the location.
- Name
- Category: The category of the location (example: Restaurant, Landmark, Event, etc)
- Address
- Comments: Your memories about the location
- Image: A url containing a link to an image for the location

The list of locations must be exported and available to use by other parts of the application.

3. Implement the server endpoints.

The server must contain these endpoints:

a. **The / endpoint**

Displays a home page that shows:

- A `hero with overlay` section.
    - Show **destination name**, **photo**, and a **brief description** about the destination.
    - DaisyUI link: https://daisyui.com/components/hero/#hero-with-overlay-image

- A `table` showing all the locations in the module you created.
    - For each location, show the image, name, category, address, comments, and a delete link.
    - DaisyUI link: https://daisyui.com/components/table/
    - Pressing the **delete button** must delete the specified location from app.

    b.   /locations/delete/[id]

- **Deletes** the location with the specified id and **redirects** the user to the / endpoint
- You are not required to handle the scenario where the provided id does not exist.

UI STYLING REQUIREMENTS

a. A sample user interface is provided in the course webpage. Your user interface does not have to be exactly the same, but it must contain similar UI elements to what is shown in the sample.

b. The app's user interface must be implemented using EJS templates.

c. Pages should be reasonably pretty and styled with Tailwind/DaisyUI.

d. Colors: Ensure you choose a DaisyUI theme and utilise DaisyUI colors (not Tailwind colors). You can choose any theme *except* "light" or "dark".

e. A fully responsive design is not required.

END OF ASSESSMENT