CSC372: Web Development
Assignment 2: Travel Blog   **Due: Thursday, September 4, 2025**

This assignment helps you practice using basic HTML and CSS. You will create a small travel blog website for a fictional traveler. The site will consist of two HTML pages and one CSS file.

File Name	Description
index.html	A homepage introducing the blog and linking to a destination travel post
destination.html	A travel blog post about a trip to a (nonfictional) destination of your choosing.
style.css	A CSS file used to style destination.html

For full credit, your files must be deployed to your public GitHub Pages repository and must match the guidelines in this document. You will have creative control over most of the content.

Index Page: Create a front page for this blog
The first part of your task is to create a front page for this blog, index.html. The front page must contain a link to destination.html. The file must also be at least 30 lines long and must contain at least 5 different HTML elements in its body. It should have significant text content. It also may not significantly borrow content from your destination.html. Otherwise, this front page can have any appearance you like. If you like, you may use an optional CSS file with this page named index.css and submit it with your other files. Be creative, have fun!

Destination Page: Create a details page for a destination of your own choosing.
The second part is to recreate a specific web page for a specific destination, destination.html. Unlike index.html, this page must exactly match the appearance specified in this document.
1. Main Heading: Travelogue: Paris (or your chosen destination).
2. Three subheadings
   a. Overview: A three-sentence paragraph to describe the destination.
   b. Must-See Spots: An ordered list of at least five items.
   c. Visitor Tips: An unordered list of at least four items.
3. A picture of the main attraction, destination1.jpg/png/jpeg. Use a caption to describe the image.
4. A testimonial about the blog (1-2 sentences) in a blockquote.
5. Link back to the home page.

Apply the following styles:
1. The page's main heading is aligned to the center of the page body and uses a 22pt bold font. Other headings on the page are left-aligned and appear in an 18pt normal font. All headings should be underlined.
2. The overall page's **body** should have a white background. Text in the body should have a foreground color of #404040 (red=64, green=64, blue=64) and use an 11pt font. The **font** families for page text are Georgia, Garamond, or any serif font available on the system. Any links on the page should use the color #A4A400 (red=164, green=164, blue=0), matching the color of the page headings.
3. The testimonial appears in an italic font as an indented blockquote with background color #FFFFA8 (red=255, green=255, blue=168). Some words in the quote are bold for emphasis.

Extra Features:

In addition to the previous required features, you must also complete at least **two** additional requirements in your destination page. These are features that may not have been covered in class; you will have to explore your resources such as online references to learn how to complete these features. If you want to complete all the extra features below, that is fine, but only two are required.

1. **Background:** Use a second image of your chosen destination as a background image for the page. The image should repeat in both directions across the page and should not move when the page is scrolled.
2. **Favicon:** Set the page to have a "favorites icon" ("favicon"). Use any image you like.
3. **List bullet:** Set the Visitor Tips list to use an image for their bullet icon rather than the normal black circle. Use any image you like.
4. **Tooltips**: When the user hovers the mouse over each of the Must-See spots, use a tooltip to show the hours for that location.

Near the top of your HTML file, **put a comment** saying which extra features you have completed.

Implementation:

For full credit, your HTML and CSS must adhere to our coding standards and validate at W3C. Choose appropriate HTML tags to match the structure of the content on the page. All CSS styles must be in style.css, do not use inline styles. Place all your images in a subfolder and use relative hrefs for them.

Format your HTML and CSS nicely so that it is as readable as possible. You must properly use whitespace and indent your HTML and CSS code. To keep line lengths manageable, do not place more than one block element on the same line or begin any block element past the 100th character on a line.