# Assignment 5: Create a Picture Gallery

## Objectives

* Write and validate an HTML file for a web browser.
* Use web-safe fonts and colors in properly written CSS.
* Use the Flexbox and Grid features of CSS.

## Instructions

Everything should go in a directory called “firstname\_lastname\_assignmentNo” (where “No” is the assignment number). No points are associated with this requirement, but if you don’t do this properly, we will deduct 10 points.

The purpose of this assignment is to build an image gallery site. The user will see a collection of images on the screen. When the user hovers their mouse over an image, it will enlarge to be the entire width of the **main** tag. This simple effect is done entirely with CSS. To do this, you’ll need images and lots of them. I downloaded ten images from unsplash.com into my **images** subfolder. Pick a theme for this gallery. I picked “mountains,” and all of my photos are about mountains, and the text is about mountains. Ten images are worth 5 points.

I wrote about 50 lines of HTML and nearly 100 lines of CSS for this assignment. So it’s a very CSS-heavy page.

### Part 1. 25 points.

In this part, you’ll be crafting your HTML. There should be four sections to your HTML document:

* header (5 points)
  + Contains a title: “Image Gallery: [Name of topic here].”
  + Contains an HR tag.
  + Contains a **nav** tag. Inside the **nav** tag put an unordered list of links to various pages about your topic. Put at least three links in an unordered list.
* main (10 points)
  + Add an “h2” element with text encouraging someone to enjoy the pictures.
  + Create a “div” element with the id “gallery”.
    - Inside of “gallery”, include all ten images. I gave each image a class name of “galleryimage”.
* aside (5 points)
  + Contains an h2 tag that outlines “About [Name of topic]” and a few sentences about your topic. I copied mine from the Wikipedia entry about mountains. :)
* footer (5 points)
  + Copyright symbol, the current year, and your name. This requirement will never change.
* I did create another div element named “twocolumn” that contains the “main” and “aside” tags, so they appear as two columns on the screen.

### Part 2

In this part, you’ll craft your CSS code. There are two instances of Flexbox and one of Grid that I want you to include.

* The body. 10 points.
  + Set the body tag to 80% width.
  + Set the margin to “auto”.
  + Use either a background image or a color other than white.
* For the h1 and h2 tags, use a Google font from the website http://fonts.google.com. I used “Dancing Script” for mine with a backup font of “cursive”. You must use a backup font. You must set the width of h1 and h2 to 100%. 10 points.
* The tags header, main, aside, and footer all have a border radius of 20 pixels and a margin of 5 pixels. Set all these tags as a “border-box” for sizing purposes. 5 points.
* The header. 5 points.
  + Make it look nice.
  + You must set the width to 100%.
* The navigation bar using the ‘nav’ tag.
  + Put a ‘ul’ tag inside the nav. 10 points.
    - Remove the list style symbol on your unordered list. The property you need to change is “list-style-type”.
    - Set the “ul” tag to be a flexbox. (This is the first flex instance.)
    - Use the “space-around” to justify the content.
  + Again, make it look nice. You are creating a professional navigational menu. Your HTML code should contain some links (and it is okay to use the “#” symbol for your links).
* Create a unique div element that we will call ‘twocolumn’.
  + This div element will contain the ‘main’ and ‘aside’ tags mentioned below.
  + This div element will appear after ‘header’ and before ‘footer’.
  + Set the display of this div to be a two-column grid. Make the grid use 75% width for the first and 25% for the second columns.
* The main tag. 10 points.
  + Again, make it look nice.
  + I’m leaving something out here involving the width of the main tag on purpose.
* The “gallery” id.
  + This will go inside the main tag.
  + Set this tag to be a flex layout.
  + Make sure this flex element will wrap the images.
  + Try to get the flex element to display three images per row of images.
  + Use a “justify-content” line to make the content look nice.
* The aside tag
  + I’m leaving something out here involving the width of the aside tag on purpose.
  + The aside tag will contain a description of whatever subject you used.
* The “galleryimage” class for the gallery images. 5 points.
  + Give each box a bit of padding (3 pixels) and a 1-pixel border.
  + Set the width to 90% of the container.
  + Set the margin to 1 pixel.
* New selector! Look up “:hover” on w3schools. 10 points.
  + When the user hovers over an image with their mouse, rotate the image 5 degrees and give it a pleasing border color.
* The footer. Five points.
  + Do I need to say it? Make it look nice.

## Turn it in

* I will deduct 20 points for every error and 2 points for every warning when validating your files. Validating your content before submitting it is critical. Each of your files should show no warnings and no errors. Here’s the validator link: https://validator.w3.org/
* **Important**: Part of web design is that your websites must be readable. Therefore, I will deduct points if anything on your page is less than 100% readable. For example, if there is text on a background and the background makes the text unreadable, I will deduct points. Likewise, if an image obstructs text flow, I will deduct points. Finally, I will deduct points if anything about your page makes it difficult to read the content.
* There are several critical parts of this assignment that (if missing) will automatically fail the project:
  + The page content isn’t as specified in this assignment.
  + Using images that aren’t taken by you or in the public domain.
  + You need to put your name in the assignment.
* You need to upload one file in the same upload:
  + **Zip up your folder** containing your HTML and image files into a single ZIP file and upload it.