

Lab Assignment



Lab Assignment

- To start off your assignment today:
 - You will do a demo



Lab Assignment

- Your assignment today is to:
 - Add a gallery page to your mini website. If you already have one, then make sure it has the same requirements. This gallery page should display at least 10 photographs. All photographs should have the same dimensions (width/height and aspect ratio). You decide how they should be arranged and display.
 - Mark up these photos using the HTML list elements. And give each photo a class attribute with value *photo1*, *photo2* and so on.
 - Then convert them to display using JavaScript arrays and loops.



Lab Assignment

- To help you in your search and preparing images for your gallery, check out these helpful tools:
 - Image search
https://www.google.com/advanced_image_search
(Recommended filters - size: *large*, usage rights: *creative common licenses*)
 - Image Editor
<https://resizeimage.net>



Lab Assignment

- With the gallery scripted with JavaScript instead of hard coded HTML, you are now ready to add some new features to this gallery.
- The new features are:
 1. A caption text below each image, like so:



BEAUTIFUL WINTER



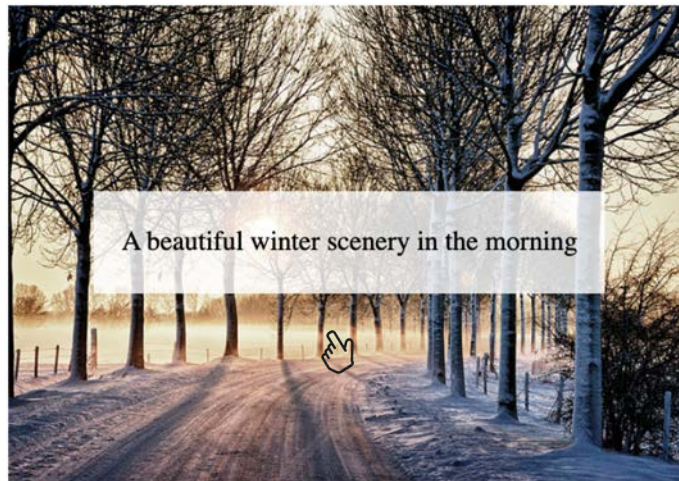
SNOW LAND

These images are only examples. You will do the same on your mini site gallery.



Lab Assignment

2. A description for each image. This description will not be visible when the page first loads. It will only be visible when the user mouse over the image:



BEAUTIFUL WINTER

This image is only an example. You will do the same on your mini site gallery.



Lab Assignment

- You will create this using your knowledge, research and problem solving skills.
- The first feature is all JavaScript. The second feature requires both JavaScript and CSS. The mouse over effect is actually done in CSS although you could do that with JavaScript as well. But most developers will tell you CSS is the better and quicker option.
- Both caption and description texts are displayed using JavaScript, similar to the gallery images. Although the description texts by default is displayed by JavaScript, the visibility of it is ultimately control using CSS.



Lab Assignment

- In JavaScript, you will need some additional variables and here is the list:
 - openCaptionTag (add a class ***caption*** on this)
 - closeCaptionTag
 - openDescTag (add a class ***description*** on this)
 - closeDescTag
- And a couple of arrays that contain a collection of texts:
 - captionTexts
 - descTexts



Lab Assignment

- In CSS (for the second feature), you will need five additional rules, targeting the:
 - list
 - caption
 - description
 - list when mouse over it (example - `?:hover`)
 - description when mouse over the list (example - `?:hover .description`)



Lab Assignment

- Some CSS essential properties you will need:
 1. **position: relative** (to set the position of the parent element (list) that contain the description box)
 2. **position: absolute** (to fixed the position of the child element (description box) with the following position properties: **left**, **bottom** to adjust the position of the description box within the list (parent)
 3. **cursor: pointer** (though optional, recommended. This makes the mouse pointer turn to a finger when above the image. Use this property on the css rule that targets the list hover ie. `?:hover.`)
 4. **visibility: hidden** or **visible** (use this property to show/hide the description box)



Lab Assignment

- One additional thing you also need to modify is the current id name on the ***openList*** tag. If you recall, in your initial HTML setup, you were asked to name the id on each list ` photo1, photo2` and so on. You will therefore need to modify the existing id name in the JavaScript variable to reflect that. To do this, you will need to modify the existing ***openList*** variables and then also, the image variable strings within the **for** loop.



Lab Assignment

- **Assignment Due:** End of Lab Session
- **Submit:** [m1-assignment.zip](#)
- **Where to submit file:** GAP - *Week 3 (Jan 18-24) Day 3 Lab Assignment*