

# Assignment #1 - Introduction to PHP

## Goal

Your goal for this assignment is to build a dynamic image gallery using [the Unsplash Image Source service](#). You will be using a PHP data-structure to store the configuration parameters for your gallery.

A working example of this assignment can be seen here: [stungeye.com/school/unsplash](https://stungeye.com/school/unsplash)

## File and Folder Structure

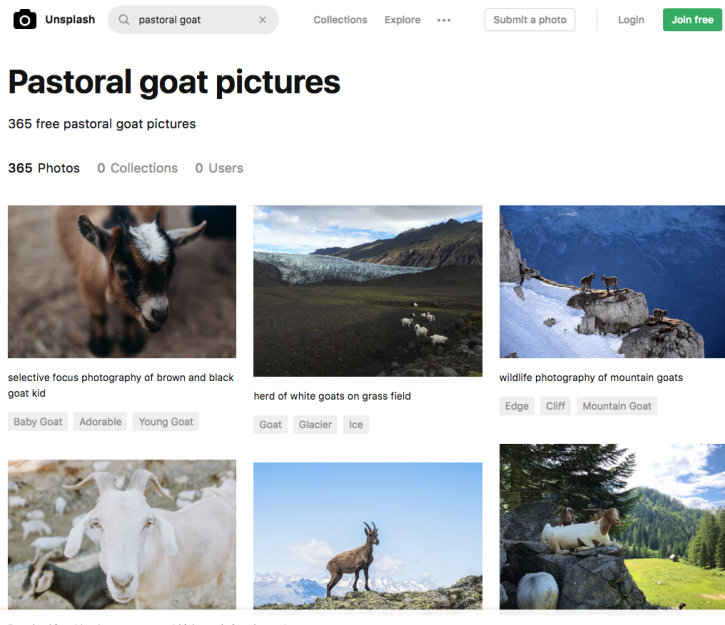
Your project requires the following files and folders:

- index.php
- main.css (Feel free to use [the one provided with the example version](#).)
- Folder named “images”

## Unsplash

[Unsplash](#) describes itself as: *“Beautiful, free photos. Gifted by the world’s most generous community of photographers.”*

Unsplash photos are searchable by keyword on their website. Here’s the website search for pastoral goats:



Unsplash also has an API for finding images by keyword using specially constructed URLs.

## Locally Saved Unsplash Images

Check out [the Unsplash website](#):

- Find and download at least four images that you like.
- Make note of the photographer's name and the photographer's Unsplash URL.
- Resize each of these images to be 1230 pixels wide (or a size that suits your CSS).  
(In MS Paint: Resize button => "pixel" radio button => Horizontal: 1230)
- Save these images to your "images" folder.

## Image Gallery Configuration

Your index.php file should be a simple valid HTML5 document with a PHP block at the top. Include the following `$config` array in this PHP block:

```
$config = [
    'gallery_name' => 'Name of Your Gallery',
    'unsplash_categories' => ['array','of','category','keywords'],
    'local_images' => ['array','of','local','image','filenames']
];
```

Update the `$config` data to customize:

- your gallery name
- the categories of images you wish to feature (minimum 4)
- the filenames of the images you saved earlier (minimum 4)

The rest of your script will use these configuration values.

## Gallery Requirements

Your gallery should include the following:

- The name of your gallery from the `$config` data should be displayed at the top of the page in an `h1` element.
- One image for each of your categories should be displayed.  
(Details on loading images by category using the Unsplash API are below. )
- Above each image, the name of the category within an `h2` element.
- A `h1` element that reads “# Large Images” where # is the size of the array of local images with the `$config` hash. (Use PHP code to find this value.)
- The local images you saved to your images folder.

IMPORTANT: You should be using loops to display all your images instead of hardcoding your image tags. Adding new categories or images should not require any changes to your code beyond the `$config` variable.

## Random Unsplash Images by Category

You can load up a random image from Unsplash from a specific category by using cleverly crafted URLs.

Loading a random 300x200 pixel “nature” photo:

```

```

## Photographer Name Links

After all of the above is working:

- For each of your local images display the name of the photographer.
- The name of display should also link to the photographer's Unsplash home page.

IMPORTANT: You need to figure out a way to store the photographer's name and URL along with the image filenames in the `$config` variable. Use your knowledge of arrays and hashes to figure out how best to store this information.

Open [the example version of this assignment](#) to see what this photographer attribution might look like.

Remember, adding new images to your page should not require any changes to your code beyond the `$config` variable.

## Wizard Tutorial - Javascript Responsive Lightbox (Optional)

A javascript lightbox allows us to view full-size images by clicking on image thumbnails.

[Here's what you'll be building](#). (Click on any of the small images.)

Let's create a lightbox gallery using the [Luminous Javascript library](#):

- Make a copy of your working `index.php` file. Call this copy: `lightbox.php`
- Modify `lightbox.php` to only display your locally stored images.
- Remove the links to the photographer's Unsplash pages.
- Make thumbnail versions of all your images that are 300 pixel wide. Name each thumbnail with `_thumbnail` at the end of the filename.

Example: `mountain.jpg` becomes `mountain_thumbnail.jpg`.

- Modify each image such that the thumbnail displayed within a link to the fullsize image. For example:

```
<a href="images/filename.jpg">
  
</a>
```

- Add these [CDN](#)-served link and script tags to your HTML head element:

```
<link rel="stylesheet"
```

```
href="https://cdnjs.cloudflare.com/ajax/libs/luminous-lightbox/2.0.1/
luminous-basic.min.css">
<script
src="https://cdnjs.cloudflare.com/ajax/libs/luminous-lightbox/2.0.1/
Luminous.min.js"></script>
```

- Add the following script tag directly *above* your `</body>` closing tag:

```
<script>
    new LuminousGallery(document.querySelectorAll(".image a"));
</script>
```

This step assumes that the CSS selector `".image a"` targets all the link tags you've got your images stored within. Modify as required, if you used different HTML from the example.

- Check if your gallery looks and acts like [the provided example](#).

## Rubric

Starting with a mark of **10**:

- Deduct **2** if the title tag and top h1 tag are missing or are hardcoded into the markup.
- Deduct **4** if the category images are missing or are hardcoded into the markup.
- Deduct **2** if the h1 above the local images is missing or contains a hardcoded number.
- Deduct **4** if the saved images are missing or are hardcoded into the markup.
- Deduct **2** if the photographer name links are missing or hardcoded.
- Deduct **3** if the HTML output of your index.php is not valid HTML5.
- Deduct **5** if the PHP generates a notice, warning or error message at any point.
- Add **2** marks if a working js lightbox was installed by following the wizard tutorial.
- If after all this the assignment mark is greater than 10, then the final mark is 10.