

IMY 220

Assignment 3:

PHP File Upload & Git

Due: Wednesday 14 August at 13:00.

The submission instructions are available on ClickUP. Any deviation from these instructions will cause a 10% deduction from your mark.

Instructions

- Use your styled registration and login functionality from Assignment 1 and 2 and add functionality for an image gallery.
- Use MySQL to add images to a gallery and display them when the user is logged in.

login.php – image upload

Download the previous assignment files (*index.html*, *style.css*, *login.php*, and *register.php*) from ClickUP. These files contain the functionality for creating and logging in as a new user using the database structure stipulated in the specification for Assignment 2. It also contains a file upload form on the login page and additional CSS for displaying images correctly. You will only need to edit *login.php* to complete this assignment.

Using phpmyadmin, create a new table in the dbuser database called tbgallery with the following rows:

- image_id: integer, primary key, auto increment
- user_id: integer
- filename: varchar(50)

Inside *login.php*, when a user selects an image with the file input, the image needs to upload and be displayed inside the user's gallery. This has to work as follows:

- When a user uploads an image (by selecting a valid file with the file input and submitting the form), the form should submit to itself (to *login.php*) and should get and upload the file selected using the file input. Use POST to submit the file.
- To do this, you will need to send through the user's email and password again. Use hidden inputs to do this.
- You should then check if the uploaded file is a jpg/jpeg image and is smaller than 1MB (you may assume that it is otherwise safe to upload). Only if the selected file satisfies these requirements should the file be uploaded to the directory called "gallery" (which you will also need to create yourself).

- A new row should also be added to the *tbgallery* table with the *user_id* of the current user and the filename that the file was uploaded with. The filename will be used to fetch the images again for the image gallery.
- The page must be error-free at all times.
- The result of this should be that a user is able to log in using *index.html* and use the file input on *login.php* to upload images. A user should be able to upload as many images as they want (one at a time).
- You do not have to cater for users uploading files with the same name. If they do, the new file will simply overwrite the old one.

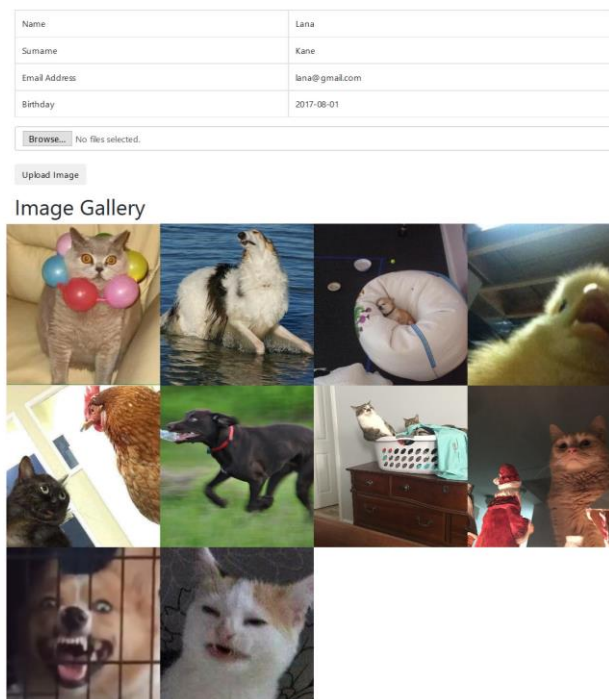
login.php – image gallery

Display all the images a user has uploaded inside a gallery. This has to work as follows:

- Beneath the table with the user's details, add a heading and a Bootstrap row. Give the Bootstrap row an additional class of *"imageGallery"*.
- Fetch all the images that the user has uploaded, in other words, all the images that have a *user_id* value of the current user's id.
- Display each of these inside Bootstrap columns. In order to display the images neatly, don't use the `` tag; instead, refer to the image file name inside the column-div's *background-image* styling. (This will require you to add inline styles.) For example, you might have several column-divs which look like this:

```
<div class='col-3' style='background-image: url(gallery/imagename.png)'>
</div>
```

Note that *style.css* includes styling rules for displaying images inside column-divs neatly, so it is important that you add the correct class to the row. The result should be that if a user has uploaded an image, their login page should display these in a gallery (including directly after they uploaded the images). For example, if a user has uploaded 10 images, their page might look like this:



Git

Create a Github account (if you don't have one already) and create a (public) repository called IMY220-Assignment3. Commit the required files to this repository and save the (public) link to this repository in a text file called *github.txt*, for example: <https://github.com/username/IMY220-Assignment3>. The result of this should be that whoever is marking your work should be able to navigate to that link, click on the Clone or Download button, and click on Download ZIP to download your file for this assignment, which is the file that will be used for marking purposes. (However, you will still have to submit *login.php* on ClickUP.)

Bonus

Allow users to upload multiple images at once. This will require you to change the upload script and the upload form created inside *login.php*. The rest of the functionality should still work as normal, in other words, all the images should be displayed in the gallery directly after they have been uploaded (no page refresh required).

Additional Information

- Refer to the slides and online resources for help

Submit only the following file(s) according to the submission instructions.

- *github.txt*
- *login.php*