Phase05 - Validation

Objectives

- Set up validation funcitons
- Add `.gitignore`
- Add a README.md file

Setup

You can use the starter-files or continue working with the code you have built.

Videos

Watch chapter 8: Validation from PHP and MySQL Essential Training: The Basics

Validation

8.1 Common data validation types

This chapter starts by listing some of the standard validation types.

Copy the validation_functions.php file into your private folder.

Read through the functions

If you have a question about how a function works, then research it using AI.

Notice that his functions return Boolean values (true or false). He doesn't actually return true or false; rather the expression is evaluated. This is a common way to write functions.

He uses a regular expression. These are incredibly powerful, but if you don't use them frequently, you will spend a lot of time looking them up (that's okay!). In most instances, PHP has a similar function to a regular expression and the PHP function executes faster.

8.2 Validate form values

As programmers, we want to validate data before creating and updating records.

Reporting all errors at once is the best user experience.

8.3 Display validation errors

Later in the video, Kevin runs into a problem and has you move three lines outside of an if...else statement. I tried using the following code to mimic the same result. Based on my testing, you do not need these lines.

8.4 Problems with validation logic

This is a short video that is worth watching for reference.

Upload phase05 to your webhost

NOTE: Make sure to change your db_credentials.php file, so the program uses your webhost's credentials. Test your program on your webhost.

Git and GitHub

Stage and commit your files.

Hiding `db_credentials.php`

We want to hide our `db_credentials.php` so people on GitHub don't see our database credentials. Use `.gitignore` to do this.

Sure! Here's a short and clear set of instructions on how to use `.gitignore` in your project:

🔒 How to Use `.gitignore` in a Git Project

1. **Create a `.gitignore` file (if it doesn't exist):**

   In your project root folder, create a file named `.gitignore`.

2. **List files or folders to ignore:**

Add each file or directory you want Git to ignore on a new line.

For example:

```
# Ignore database credentials
private/db_credentials.php

# Ignore all log files
*.log

# Ignore folders
/node_modules/
/vendor/
```

3. **Make sure the file is not already tracked:**

   If the file was committed before, remove it from Git tracking:

   ```
   git rm --cached path/to/your/file
   ```

4. **Commit your changes:**

   ```
   git add .gitignore
   git commit -m "Update .gitignore to exclude sensitive or unnecessary fi
   git push
   ```

5. **Done!**
   Git will now ignore the listed files and folders.

Test your work by going to GitHub and make sure the `db_credentials.php` is not longer available.

Create a README.md file

We want to add instructions so people coming to our site can recreate the `db_credentials.php` file.

Create a file named `README.md` and store it in your project root folder.

README.md files use Markdown which is an HTML preprocessing language. It makes writing HTML extemeley easy.

I use DaringFireball as Markdown source, but there are many others.

You can use this README.md file from my site as an example. You also modify it.

Great — based on your info, here's a clean `README.md` you can use for your PHP/MySQL project:

PHP/MySQL Project

This is a basic PHP project that connects to a MySQL database using credentials stored in a secure, non-tracked file.

Requirements

- PHP 7.4+
- MySQL 5.7+
- Web server (e.g., Apache, Nginx)

Installation

1. **Clone the repository:**

```
git clone https://github.com/your-username/your-repo-name.git
cd your-repo-name
```

2. **Create a `db_credentials.php` file:**

Inside the `private/` folder (create it if it doesn't exist), create a file named `db_credentials.php` with the following content:

```
<?php
define("DB_SERVER", "");
define("DB_USER", "");
define("DB_PASS", "");
define("DB_NAME", "");
```

Fill in the appropriate values for your MySQL setup.

1. **Protect your credentials:**

   Make sure `db_credentials.php` is listed in your `.gitignore` file so it is not committed to version control:

   ```
   /private/db_credentials.php
   ```

Usage

The program is written so the `initialize.php` will always include the `db_credentials.php` file.

Security Note

Never commit `db_credentials.php` to version control. Use `.gitignore` to prevent accidental exposure of sensitive credentials.

Submit your work

- Submit your GitHub and webhost addresses in the comments section of Moodle.
- I have left open the link to submit your zipped files in case you are having difficulty uploading to GitHub.