COMP 3512 - Group Tutorial #2[1] – February 6, 2026

Due on February 6, 2026

Goals

In this tutorial, you will learn how to work with PHP arrays and superglobals in form processing. You will also review loops, file inclusion, and CSS.

For this tutorial, please **disable AI assistance** in VS Code or Codespaces. These exercises are intended as practice for your exam, which will be written on paper and will not allow the use of AI tools. To disable AI assistance, follow the steps below:

- Open **VS Code** or **Codespaces**.
- Go to **File > Preferences > Settings** (or press **Ctrl + ,**).
- Search for **"disable AI"**.
- Locate **Chat** and disable settings related to **GitHub Copilot Chat**.
- You may need to refresh your Codespaces page or restart VS Code for the changes to take effect.

The required files and starter code have been provided on **D2L**.

For this tutorial, you may need to review some **HTML tags**. If you don't remember them, you can quickly look them up online. The HTML tags needed for this tutorial include:

- <select> and <option>
- and
- <form> elements, including form attributes and input fields
- <table>, <tr>, and <td>

[1] This tutorial has been adapted and modified from Project 2 in Chapter 12 of the textbook, "Connolly and Hoar, Fundamentals of Web Development (Pearson, 2022)".

Group Work

In this tutorial, you will practice form processing and using superglobals and arrays in groups of two students. One of the main purposes of group tutorials is to help you learn how to collaborate on a project, resolve potential conflicts, and learn from and teach each other.

You may choose one of the following approaches:

- **Work on the entire assignment together**, which is recommended if you have questions or uncertainties about how to proceed. This approach allows you to discuss ideas, clarify concepts, and learn from one another.

- **Divide the tasks** (for example, implementing different functions) between group members and then combine your work. This approach is effective when both members clearly understand the assignment requirements and can work independently on smaller components.

You may also use a **hybrid approach**: begin by working together to understand the requirements, and once both group members feel confident about the implementation, divide the tasks. Choose the approach that best supports effective collaboration and learning for both group members.

This lab is expected to take approximately **40–50 minutes** to complete in a group setting.

You will receive a full, half, or zero mark depending on how much effort you put in and how much work you complete.

Submission

At the end of the tutorial, share your work with me via GitHub or submit it through D2L. Please include the names of both group members as a comment at the top of your main php files.

Instructions

1. You have been provided with two files: the data entry form (`ch12-proj21.php`) and the file that processes the form data (`art-process.php`). Open both files in your browser and examine their output.
2. Modify `ch12-proj21.php` to use the **POST** method and set `art-process.php` as the form's action.
3. In `header.inc.php`, write a loop that uses the `$links` array from `data.inc.php` to generate a list of hyperlinks. Then include `header.inc.php` in both `ch12-proj21.php` and `art-process.php` immediately after the `<body>` tag. This will create a header with a menu list on both pages of your website. The expected output for this step is shown in **Figure 1** and **Figure 2**.

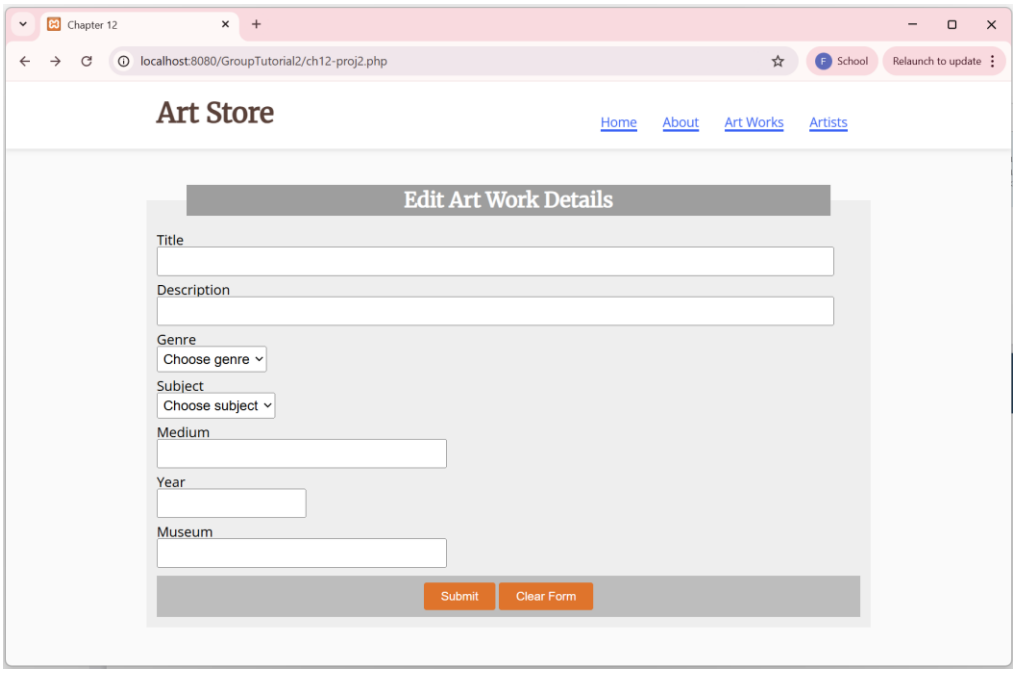

Figure 1: ch12-proj21.php output for Exercise 3

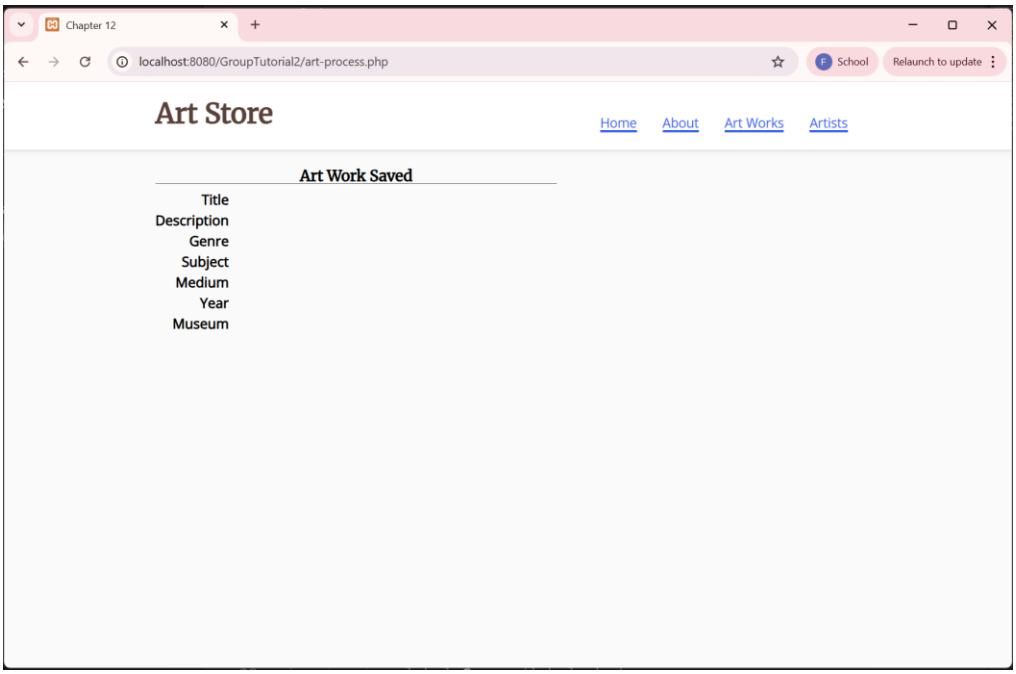

Figure 2: art-process output for Exercise 3

4. In `data.inc.php`, define two string arrays: one for the genres **Abstract**, **Baroque**, **Gothic**, and **Renaissance**, and another for the subjects **Animals**, **Landscape**, and **People**.

5. Write a function that accepts a string array and returns a string containing each array element wrapped in an `<option>` tag. Use this function to generate the **Genre** and **Subject** `<select>` lists.
Hint: Review the function-related topics in the **PHP Fundamentals 2** lecture slides for guidance on defining functions.

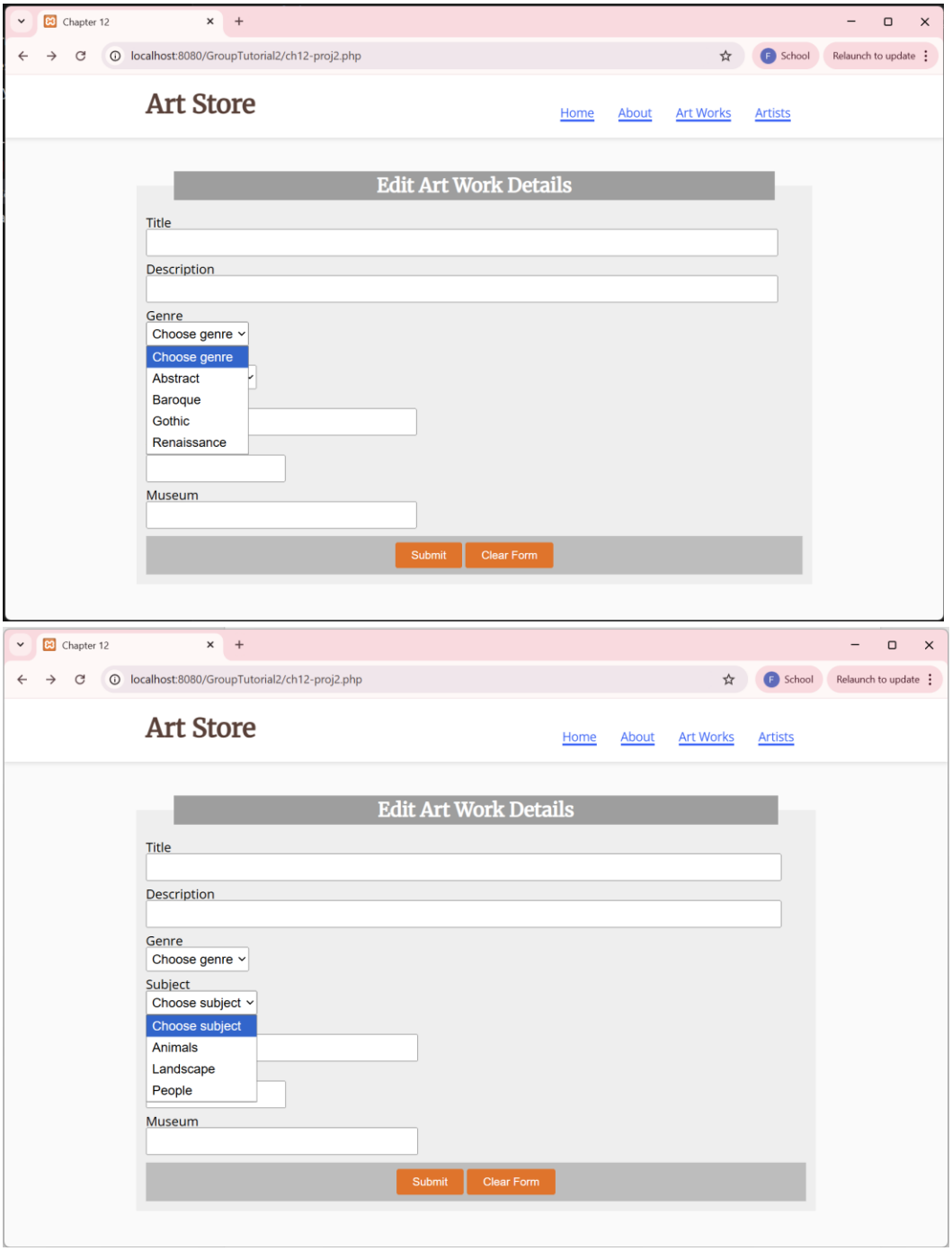

Figure 3: Output for Exercise 5

6. Modify art-process.php to display all values submitted through the form, as shown in **Figure 4**. This will require using the appropriate **superglobal array**.
Hint: Refer to **slide 8** of the superglobals lecture.

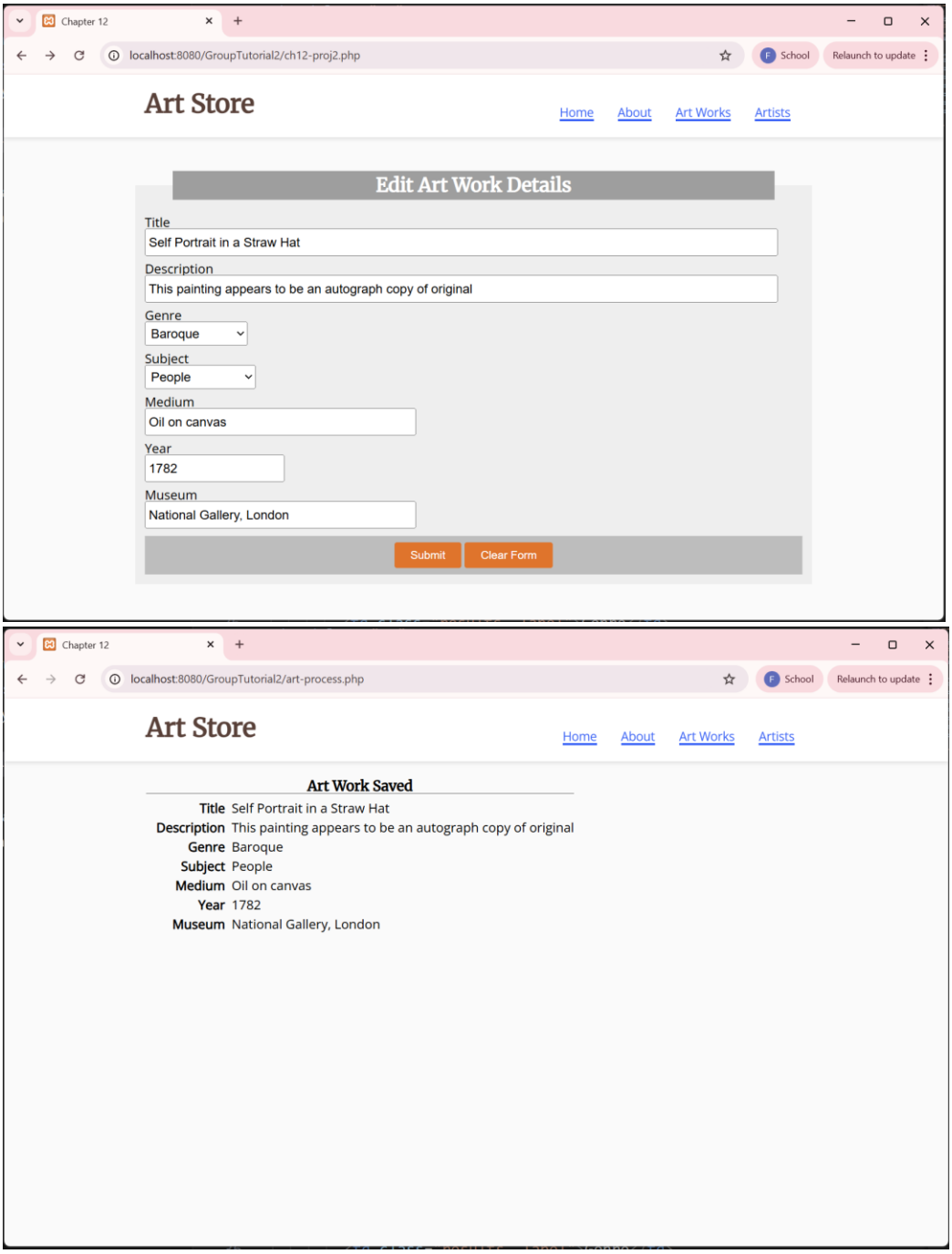

Figure 4: Output for Exercise 6

7. Ensure that appropriate error message is displayed when art-process.php is accessed **without POST data**. This occurs when the page is opened directly (for example, by typing the URL in the address bar and pressing **Enter**, or by refreshing the page) rather than by clicking the **Submit** button. In this case, the request method will be **GET**, not **POST**. The error message should also

include a hyperlink that allows the user to return to the ch12-proj2.php page.
Hint: Use the $_SERVER superglobal to check whether the request method is POST. Refer to the **Superglobals** lecture slides for guidance.

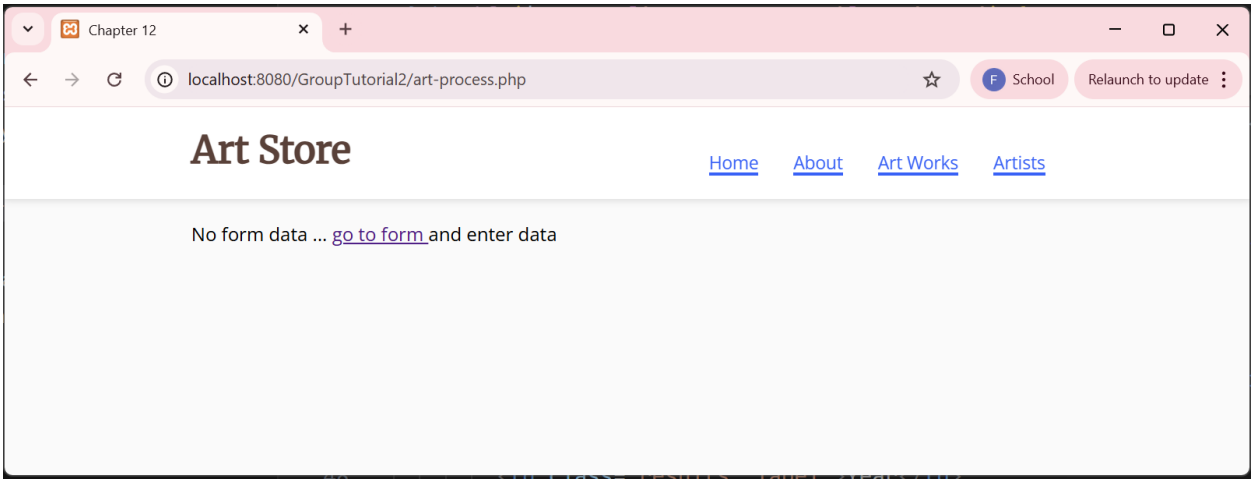

Figure 5: Output for Exercise 7

8. Make the **Title** field required. If the user leaves this field empty or enters only whitespace characters, the form data should not be processed. Instead, display an appropriate message informing the user of the issue, along with a hyperlink (like the hyperlink generated in Exercise 7) that allows them to return to the form page. *Hint:* Refer to **slide 9** of the Superglobals lecture slides.