Darliza Sanchez
10/22/24

Project #1

Book club

The Mist Book Club Platform is a user-friendly setting designed for anyone who wish to participate in a vibrant book club community. Book enthusiasts may interact, connect, and discover a variety of literary experiences on the platform.

Key Features of the Platform:

Joining Details: The website gives a full overview of what it will take to be a member, the requirements involved in becoming one, and how one can easily register to join the Mist Book Club. Book Catalog: The members are given a selected list of the books that the club has read and the novels under consideration. Events Schedule: Events scheduled for upcoming meetings, discussions, and events are posted on the site. One will easily know when the next meeting is held, both physically and electronically. Member Profiles: Club member profiles are provided, which gives way to relationships and a community feeling. Contact and support: Easy contacting of the club organizers by website users is made easy on a dedicated contact page for questions, recommendations, and other information. Attractive Design: The platform is warm, friendly, and appealing to the eyes. As a result, users may find it easy to navigate through this website and find whatever they are looking for without burdening themselves.

Project1index.php

The purpose of Project1index.php is to serve as the homepage for the "Mist Book Club" website. It gives guests an overview and a warm greeting, urging them to explore the club and take part in its events. With a kind greeting and an appropriate image, the page is meant to be straightforward but friendly. The structure of the page is pretty regular because it uses common terms just like any other HTML document. It contains: A DOCTYPE declaration to denote the version of HTML. A <head> section providing a linkage for the page title and its styles. A <body> section with three basic things inside: a header, a welcome message, and an image. The header and footer of the website for the book club are dynamically included in this page using this PHP include function. This eliminates any chances of making an error in other pages and preserves design consistency across several pages. The major challenge was the consistency across multiple pages and maintenance of responsive design, which was handled through modular PHP code and responsive image styling. It makes this page very simple and easy for new users to enter inside and start navigating.

Project1about.php

The Project1about.php page of the Mist Book Club website informs every visitor about the club or, more precisely, about the mission and values that support it. It shows basic information about the philosophy, advantages derived from it, and also what a member can expect while participating in this club. Just like the index, the about page has uses common terms just like any other HTML document. It contains: A DOCTYPE declaration to denote the version of HTML. A <head> section providing a linkage for the page title and its styles. A <body> section with the club's information. Main content goes into the <section> element. The section opens with a heading entitled "About Us" and then several paragraphs of text that describe how the Mist Book Club was started, its mission, and also the values of the community.

One of the tasks of creating content able to reach the targeted audience was necessary. All this had to do with warmth, invitation, and reflection upon the all-inclusive nature of the club. Words were softly to be chosen to appeal to the targeted potential members by pointing at values in the club that deal with communities. A major addition to this page is the content focused on community building and shared experiences. The idea is to highlight the difference between Mist Book Club and just any old reading club, emphasizing values of connection, debate, and support. In that sense, this sets up the mission of the club and invites new users. The page was designed very simple and consistent with the rest of the website. The content is given in small paragraphs, so it does not overload the visitor's eyes with a large amount of text all at once.

Project1processorder.php

This script processes form submissions coming from the book club site. It should take two kinds of forms: join form for people joining the club, and contact form for sending messages to someone. It writes the information after it processes the form to a file and displays a confirmation message.

How It Works: Using the example code given in chapter 2 the code starts by the Opening of HTML page: the script opens an HTML page by giving a title to the page and styles, and it also provides the website's header. Form is submitted check: The script checks if any form is sent to the server through user action via the POST method; that is when a user submits a form on the website upon clicking "Submit". Which form was submitted: Then, the script has checked which form has been sent: If the form sent was the join form with full name and favorite book genre, it continued to process the user's full name, email, and favorite book genre. Or if it was the contact form (with name and message), it processed the user's name, email, and his message. It includes a thank-you message that says the user has joined the book club, provided the join form was submitted. In case the contact form was submitted, it displays a thank-you message for reaching out and displays their message. Finally, having processed the form, the script then prepares the submission data-like name, email, and message-and saves it in a text file called newmembers.txt. This would serve to keep track of all form submissions. If there's a problem saving the submission to the file, the script shows a message saying the submission couldn't be processed at the moment.

processorder.php from chapter 2 and Project1processorder.php, Both scripts process form data submitted via a POST request, sanitize the inputs, and store the results in a text file. They both append the submitted data to a file (using fopen(), fwrite(), and file locking with flock().

The main Challenge Faced was Handling Different Form Types in One Script. Project1processorder.php needed to process both the join form and contact form. This was resolved by checking which fields are set in the POST request to differentiate between the two forms (isset($_POST['fullname']) for join form, isset($_POST['name']) for contact form). I added some line comments in the code itself to make it a bit clear where everything happens.

Project1members.php

Project1members.php is a part of a book club website that displays a list of members along with their favorite books. The page shows a header, a list of members, and a placeholder image. The script creates an HTML webpage that shows the Members. In addition to that, it connects to a CSS file for styling. The page has a header, which it draws from a different file named Project1header.php. The page has a section for "Our Members" and shows an image related to the book club (BK.jpg). The image is designed to be round with a border which i had to look up how to do because in the examples provided in the class was just squares images. The script uses an array named $membersArray to list the different members. Here's how it's structured:

"Member Name : Favorite Book Book Title by Author"

The code then loops through this array and displays each member's name and favorite book as a list. The page also includes a footer by loading the Project1footer.php file. There is a code in chapter 3.8 that defines an array of image file names (representing different car parts) and uses shuffle($pictures); to randomize the order of these pictures each time the page is loaded. I used the function shuffle($membersArray); to randomize the names of the members and added the code in order to make the page more interactive every time it gets loaded, showing the names in a different order. Displaying the members in a neat format, and keeping the structure the same throughout, is one challenge. Using an array ($membersArray) to hold the member names and favorite books was an easy, neat solution. Looping through this array, the page displays the information of each member in a neat fashion. I added some line comments in the code itself to make it a bit clear where everything happens.

Project1join.php

Project1join.php is a form to join the Mist Book Club for new members. The major objective of this script is to gather information from users who would want to sign up for the club, such as name, email address, and favorite book genre. This information is then submitted for processing to Project1processorder.php. Overview How It Works- The page simply has a title along with links to a CSS named styles.css. This CSS is used for styling forms and text mainly. It pulls in the header file, named Project1header.php that helps in maintaining consistency across the website. This page calls on the user to actually enroll in the book club. It introduces information about what the club offers such as meeting times, location options-in person and virtual-and an overview of the club's structure, all done with a simple print line: <p> </p>. The principal attraction of this page is the form collecting information for new members. Here is the breakdown of what the form will include: Full Name, Email Address, Genre. Dropdown list where user can choose their favorite book genre. User can select between fiction, non-fiction, mystery, fantasy, and biography. Upon completion, the form is submitted to Project1processorder.php for processing. The page also contains a footer file: Project1footer.php.

Some challenges creating this page included: Form Validation and the drop-down list. The forms should check their input for data quality. For example, the email is in the right format and all fields have an entry before submitting the form. I used the 'required' attribute in the form fields so that a user is compelled to fill in the information required for them to submit the form. This basic form validation of empty fields. First, to get the dropdown list working, I created the label for the dropdown list by using the <label> element. This ensures that the dropdown is correctly labeled, such that a user would know what they are selecting. This attaches to the dropdown list by setting the for attribute to the ID of the dropdown element- favorite_genre. The actual dropdown is created using the <select> that contains a few options from which to choose by the user. An id and name attribute ensure that the data coming out of this dropdown is submitted correctly and is tied to the script that processes this form. This id attribute refers this dropdown to this label directly above it. The name attribute names the field, which will be utilized if the form information is submitted. In this case, the required attribute prevents the user from submitting the form without making a selection.

Project1Contact.php

Project1Contact.php allows users of "The Mist Book Club" to contact them using a contact form. It collects basic contact information: the user's name, email, and message; and then submits this information to Project1processorder.php for processing. <form action="Project1processorder.php" method="post">. Inside the form, there is a "Send Message" button that the user clicks to submit the data: <input type="submit" value="Send Message">. The page also holds a header, Project1header.php, and a footer, Project1footer.php, which maintain the same navigation and styling across the site.

For this page i followed the code given in chapter 4, Processfeedback.php. The processfeedback.php script is designed to handle feedback from a customer on a different website "Bob's Auto Parts". It receives the user's name, email, and feedback then sends that on via email to a predefined email address feedback@example.com. It handles feedback for the site owner. A major edit made was whereas in processfeedback.php, the form data is mailed to a pre-set address. In Project1Contact.php, no such emailing within the form processing was made. The form data in the latter file gets forwarded to Project1processorder.php.

Project1Books.php

The purpose of the Project1books.php page is to list books that the Mist Book Club has or will select for current and past months. Any visitor would get an idea as to what books the club was reading over time and get a view of the nature of books taken up for discussion in the club. This page is a must so that members and visitors know what is being read currently and what has been chosen previously. This page starts with the structure of an HTML document including a title: Mist Book Club - Books and the link to the CSS stylesheet so this page looks the same as the others. The PHP include statements include the header and footer. These are pieces of code reused throughout the site to create consistency in layout and navigation. The body of this page consists of a heading, Current and Past Selections, along with a list (<ul>) that displays the book titles. The book titles and months are stored in a multidimensional array called $booksArray.

The try-catch block is used in this PHP script to handle potential issues, such as if the $booksArray is empty. Chapter 7 uses a try-catch block in finally.php to display an error message rather than a broken page. I do not think that Project1Books.php necessarily needed it but because I could be working with potentially problematic data in the future, this is a smart practice. Inside the try block: It tests whether the $booksArray is empty. If it is, an exception is thrown with an "No books available." Message. If it is not an empty array, it iterates an array using a foreach loop, showing each book in the array as a list item. The finally block is executed regardless of whether an error occurred or did not occur. It prints a closing message, thanking visitors for visiting to review selections.

One might say that a potential challenge here in the code is just ensuring the page doesn't break if something goes wrong-whether it is an empty book list or an issue with data. A try-catch block has been used to manage this problem in order to catch any exceptions and provide the user relevant error messages if any do occur. Every book's title is stored in the array and is dynamically displayed on the page. This gives great ease to any management of the list of books, ensuring that one need only alter the array rather than the HTML structure directly. Finally block makes sure that no matter what happens in the processing of the script, a friendly message thanking user for checking selections of books will be shown.

Project1Header

Project1header.php is the file that allows the view to have the same header and the same navigation menu across the website; it keeps the consistency in layout on the website and for the user experience by calling this file in other pages through the 'include' function of PHP. This gives to the web application the possibility of reusing components, or elements that could easily maintain the structure of the site to update or modify it once rather than change it on every page. A clear designation of the club's name, "Mist Book Club," is located at the top of the website along with an image in the header. Picture would be designed as a circular logo on setting CSS parameters like object-fit: cover, border-radius: 50%. This makes it sure that picture fits in his limits without any distortion. This would be the title of the book club, styled to be the main focal point of the header.

Chapter 5 and homework 2 play a main part in creating the headerThe Navigation menu, which links the major section of this Website, from Home to About, Books, Members, Join and Contact. Each link is visually represented with an icon here bcMenu.jpg, followed by associated menu text. Each of the navigations is wrapped in a div classed menuitem. Each div contains an anchor tag linked to a corresponding page: Home, About, Books, Members, Join, Contact. Each link includes a small icon image - BCMenu.jpg - along with the menu text itself. All icons and menu items are of the same dimension, 20x20 pixels for the images, thereby maintaining consistency in the appeal. The other challenges were how to make the logo image appear in a circular shape without distortion and how to keep the items of navigation well-spaced and mesmerizing to see. The Solution: The use of CSS properties, such as border-radius and object-fit, helped solve this by setting regular dimensions for the images. Also, placing the navigation inside a div element made working with the menu positions and spacing much easier.

Project1Footer + Project1Legal

The Project1footer.php file contains the footer section of the Mist Book Club. As in the case of the header, it is created to be standardized-that is, to have a standard header across all the pages of the website. In such a case, this footer can then be used throughout several pages with PHP's include, while the content of the footer remains the same throughout the site. It provides general information like copyright and a link to the legal info page. The footer contains a copyright notice and a link to the website's legal information page. The footer will carry a dynamic copyright notice for the year 2024. It tells the users that the website and its contents are under the copyright laws. &copy; is the HTML entity for the copyright symbol, which shows "©". The text contains the name of the club followed by the year of protection "2024."

The footer provides a link to the legal information page: Project1legal.php. This can include but is not limited to terms of service, privacy policy, disclaimers, and other legal details relevant to the users of this site. The legal link, therefore, forms one of the best practices for websites, those that show transparency in how it deals with user data and what a user has agreed to when he uses the site, among other legal obligations. All of this is from Chapter 5 and homework 2 and I followed its guide in order to complete both the footer and legal page.

One of the challenges when working on creating a website footer is ensuring that the footer content is consistently reflected on all pages. Without a reusable structure in place, you might have to tediously copy the footer across each page, which can lead to inconsistencies or even errors. This is done by creating Project1footer.php and using PHP's include function; this allows the footer to be added to every page without code duplication. Other considerations were how to treat the link to the legal information page in such a way that would make users easily proceed to legal content without polluting the footer with sentences. It is just an HTML <a> element. That makes things clear and accessible and the user will get any legal information he needs without loading the footer with text.

Styles.css

Last but not least is the styles.css. The CSS file is responsible for setting the visual appearance and layout in the Mist Book Club website. CSS controls the styling of the HTML elements: the colors, the fonts, spacing, positioning, among others. The goal of the styles.css is so that all the pages of the Mist Book Club website look the same and appealing. In this CSS file, the color palette is specified, such as backgrounds and text colors, among other embellishments. I've selected a few colors for backgrounds and text with the intention of fitting into the style of the Mist Book Club. While most of the base code was taken from Chapter 6, the color and font schemes have been changed to best fit this website for the book club. Problems that arose from choosing appropriate colors, fonts, and assurance of consistency in layout were overcome through careful testing and adjustments.

The Mist Book Club Platform is designed in a way that makes it easy for users to go through it. It has a very transparent menu that guides the users through other sections like About Us, Home, Books, Members, Join, and Contact Information. Each page is at once cognitive and entertaining, encouraging the active participation and interaction of its members. Generally, the Mist Book Club Platform is an insightful online space that keeps prospective and current members alike well-informed about the club in general, while at the same time providing a community feel to the features themselves and even to the ease of navigation. Every page invites people into activity and connection by its fun and instructive manner. This means that, in general, the Mist Book Club Platform is an informative online tool that can help both potential and existing members learn about the club in general, while giving a sense to the community among the features and even navigation.