

5. Describe best practices for using persistent cookies.
6. What is web storage in HTML5? How does it differ from HTTP cookies?
7. What is session state?
8. Describe how session state works.
9. In PHP, how are sessions stored between requests?
10. How does object serialization relate to stored sessions in PHP?
11. What is a web farm? What issues do they create for session state management?
12. What is caching in the context of web applications? What benefit does it provide?
13. What is the difference between page output caching and application data caching?

16.9.3 Hands-On Practice

PROJECT 1: CRM Admin

DIFFICULTY LEVEL: Basic

Overview

Demonstrate your ability to work with Cookies in PHP. You will create and read both persistent and session cookies, as shown in Figure 16.15.

Instructions

1. You have been provided with a starting file named `chapter16-project1.php` along with a second page named `other-page.php`. The first file will be used to create the cookies as well as read them; the second page will verify that the cookies are available across other pages in the same domain. Examine the `<form>` element in `chapter16-project1.php` and note that the action is a file named `make-cookies.php`.
2. Create a new file named `make-cookies.php`. This file will contain no markup: it will just save the form data (the values of the two `<select>` lists) as cookie values.
3. After checking for the existence of the relevant form data, save the theme value as a persistent cookie using the `setcookie()` function. Set the expiry to be a day from the current time. You may need to set the domain value, which is the fifth parameter to the `setcookie()` function. Save the philosopher value as a session cookie by setting the expiry to 0. After setting the cookies, redirect back to `chapter16-project1.php` using a `header("Location: chapter16-project1.php")` function call.
4. Within the “Reading the Cookie” card in `chapter16-project1.php`, read and display the contents of these two cookies. Be sure to display an appropriate message of the cookies are not available (see Figure 16.15).



**HANDS-ON
EXERCISES**
Project 16.1

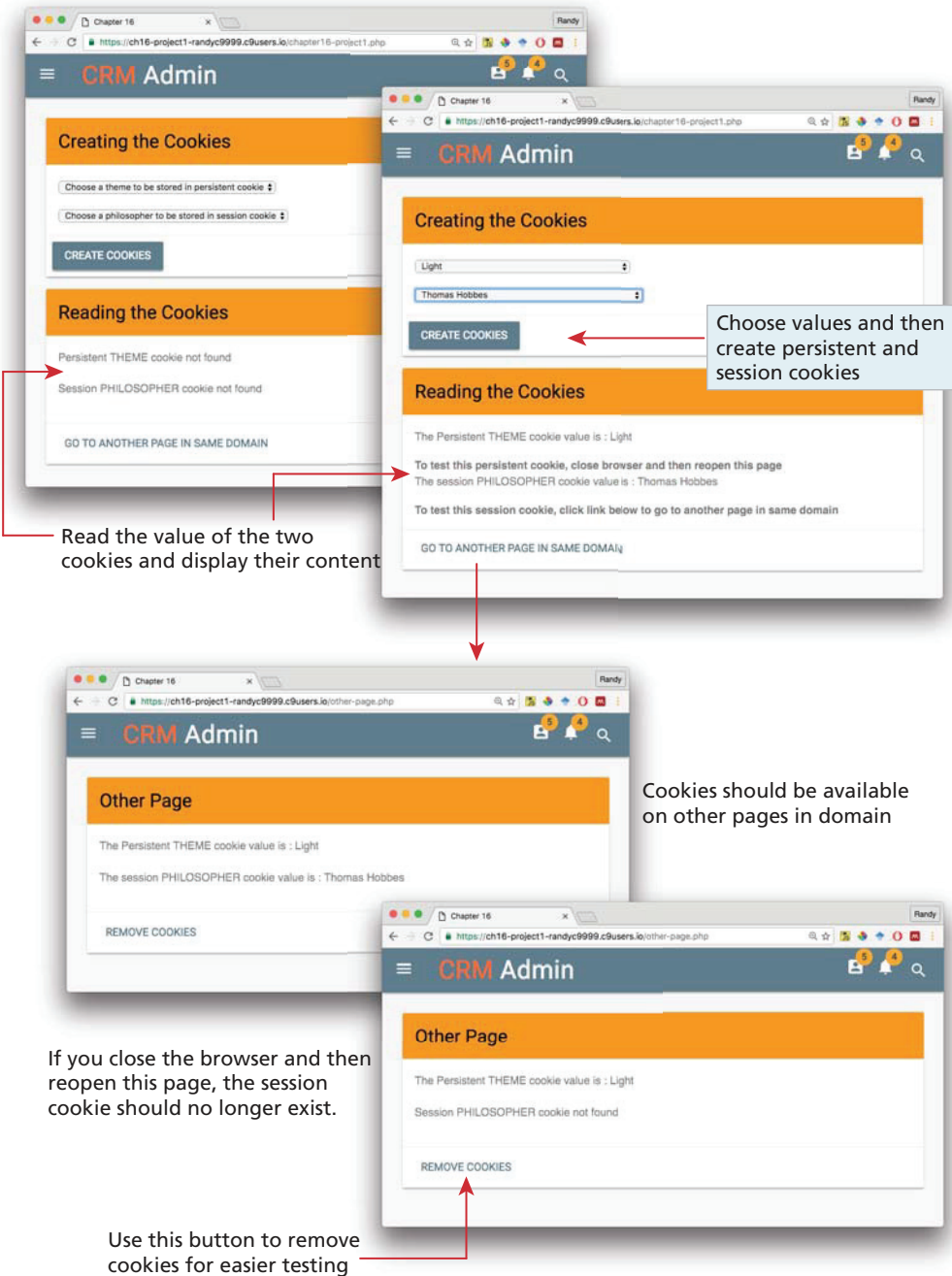


FIGURE 16.15 Completed Project 1

5. Add the same read and display cookie code to `other-page.php`. Notice that the link for Remove Cookies is for a file named `remove-cookie.php`.
6. Create a new file named `remove-cookie.php`. This file will contain no markup: it will just remove the cookies. To do this, use the `unset()` function on the two cookie values within the `$_COOKIES` array. As well, use the `setcookie()` function but with an expiry date in the past. Afterwards, redirect to `chapter16-project1.php`.

Test

1. You may need to close the browser entirely to test your session cookies.

PROJECT 2: Art Store

DIFFICULTY LEVEL: Intermediate

Overview

Building on the PHP pages already created in earlier chapters, you will add the functionality to implement a favorite paintings list using a session variable, as shown in Figure 16.16.

Instructions

1. Begin by finding the project folder you have created for the Art Store. Session integration requires adding the `session_start()` function call to all pages that will use session data.
2. Both `browse-painting.php` and `single-painting.php` contain Add to Favorites links styled as buttons. Modify these links so that clicking on them will take the user to `addToFavorites.php`. These links need to provide indicate which painting to add to the favorites list via a query string. To make our view favorites page easier to implement, include the `PaintingID`, `ImageFileName`, and `Title` fields in the query string.
3. Create a new blank page, `addToFavorites.php`, which will handle a GET request to add a painting to the favorites list. This file will contain no markup: it will check for the existence of the relevant query string fields, and then add the painting information to session state.
4. The favorites list will be represented as an array of arrays. Each favorite item will be an array that contains the `PaintingID`, `ImageFileName`, and `Title` fields for the painting. You will need to retrieve the favorites array from session state (or create it as a blank array if it doesn't exist), and then add the array for the new favorite item to the favorites array. You must then store the modified favorites array back in session state. After this, redirect to `view-favorites.php` using the `header()` function.
5. Modify the `view-favorites.php` page so that it displays the content of the favorites list in a table. For each painting in the favorites list, display a small version of the painting (from the `images/art/works/small-square` folder) and its title. Make the title a link to `single-painting.php` with the appropriate querystring.



**HANDS-ON
EXERCISES**

Project 16.2