



**FIGURE 12.15** Completed Project 1

#### Test

1. Test the page. Remember that you cannot simply open a local PHP page in the browser using its open command. Instead you must have the browser request the page from a server. If you are using a local server such as XAMMP, the file must exist within the `htdocs` folder of the server, and then the request will be `localhost/some-path/Chapter12-project1.php`.

## PROJECT 2: Share Your Travel Photos

**DIFFICULTY LEVEL:** Intermediate

#### Overview

You have been provided with two files: a page that will eventually contain thumbnails for a variety of travel images (`list.php`) and a page that will eventually display the details of a single travel image (`detail.php`). Clicking a thumbnail in the first file will take you to the second page where you will be able to see details for that image, as shown in Figure 12.16.

#### Instructions

1. Both pages will make use of arrays that are contained within the include file `travel-data.inc.php`. Include this file in both pages.
2. Both pages display a list of countries. Replace the hard-coded lists by looping through the `$countries` array to display a list (in `details.php`, the list is contained within the include file `left.inc.php`). Be sure to first use a PHP sort



**HANDS-ON EXERCISES**

Project 12.2

**Write a loop to display countries using an array. Each of these is a link to list.php with the country as a query string.**

**Each of these images will be a link to detail.php with the id of the image passed as query string.**

**Write a loop that displays these images and links using data within the \$images array.**

**Display the appropriate data from the \$images array.**

**Write loops to display these lists. Also use the appropriate PHP sort functions.**

FIGURE 12.16 Completed Project 2

function. Each country in the list should be a link to `list.php` with the country name as a query string parameter. Also replace the continents hard-coded list with a loop as well.

- In `list.php`, replace the existing image list markup with a loop that displays the thumbnail image and link for each of the elements within the `$images` array (which is provided within `travel-data.inc.php`). Notice that the links are to `detail.php` and that they pass the `id` element as a query string parameter.

4. After testing `list.php` to verify it works as expected, add logic to handle the country links. Each link in the country list should be to `list.php` but with the country name as a query string (e.g., `list.php?country=Canada`). You will need to filter the images list so that the page displays only those images from the specified country.
5. In `detail.php`, retrieve the passed `id` in the query string, and use it as an index into the `$images` array. With that index, you can output the relevant title, image (in the `images/travel/medium` folder), user name, country, city, description, and tags.

#### Test

1. Test the pages in the browser (see the test section of the previous section to remind yourself about how to do this).

### PROJECT 3: CRM Admin

DIFFICULTY LEVEL: Advanced

#### Overview

Demonstrate your ability to fill arrays from text files and then display the content.

#### Instructions

1. You have been provided with a PHP file (`Chapter12-project3.php`) that includes all the necessary markup. You have also been provided with two text files: `customers.txt` and `orders.txt` that contain information on customers and their orders.
2. Read the data in `customers.txt` into an array, and then display the customer data in a table. Each line in the file contains the following information: customer id, first name, last name, email, university, address, city, state, country, zip/postal, phone, and sales. Each of these fields is delimited by semicolons. You will notice that you are only displaying some of that data.
3. Each customer name must be a link back to `Chapter12-project3.php`, but with the customer id data as a query string (see Figure 12.17).
4. When the user clicks on the customer name (that is, makes a request to the same page but with the customer id passed as a query string), then display additional customer information in the Customer Details card. Also read the data in `orders.txt` into an array, and then display any matching order data for that customer (see Figure 12.17). Each line in the orders file contains the following data: order id, customer id, book ISBN, book title, and book category. Be sure to display a message when there is no order information for the requested customer.
5. The sales field in the customers table is a series of 12 comma-separated numbers. You will use sparklines.js jQuery library to display those numbers as an inline bar chart. Examine the sample customer table row to see how easy it is to make this data look impressive using jQuery!



HANDS-ON  
EXERCISES  
Project 12.3