Chapter 12

How to work with cookies and sessions

Objectives

Applied

1. Use any of the functions and techniques presented in this chapter as you use cookies and session tracking in your applications.

Knowledge

1. Describe the use of cookies, and distinguish between per-session and persistent cookies.
2. Describe the use of session tracking.
3. Describe the use of the $_COOKIE and $_SESSION variables.
4. Describe the use of the functions for working with cookies and sessions.

Examples of cookies

```
PHPSESSID=D1F15245171203E8670487F020544490
user_id=87
email=jsmith@hotmail.com
userName=jsmith
passwordCookie=opensesame
```

Terms

- cookie
- per-session cookie
- persistent cookie

How cookies work

- A cookie is a name/value pair that is stored in a browser.
- On the server, a web application creates a cookie and sends it to the browser.
- On the client, the browser saves the cookie and sends it back to the server every time it accesses a page from that server.
- By default, cookies only last until the user closes his or her web browser. However, cookies can be set to persist in the user's browser for up to three years.
- Some users disable cookies in their browsers.
- Browsers generally accept only 20 cookies from each site and 300 cookies total.
- Browsers can also limit each cookie to 4 kilobytes.

The syntax of the setcookie function

```
setcookie($name, $value, $expire, $path,
          [$domain, $secure, $httponly])
```

Setting a cookie in the browser

```
$name = 'userid';
$value = 'rharris';
$expire = strtotime('+1 year');
$path = '/';
setcookie($name, $value, $expire, $path);
```

The setcookie parameters:

- $expire: default = 0; lasts until user closes browser window; a *per-session* cookie. Other timestamp values are *persistent* cookies.
- $path: The server path the cookie is available to. If it is set to '/', the cookie is available to all directories on the server. The default value is the directory of the file setting the cookie.

Getting the value of a cookie from the browser

```
$userid = $_COOKIE['userid'];  // $userid is 'rharris'
```

$_COOKIE is an associative array, key->value where key is the cookie name and value is the cookie value.

Deleting a cookie from the browser

```
$expire = strtotime('-1 year');
setcookie('userid', '', $expire, '/');
```

The setcookie function must be called before any HTML output is sent in the response.

How to enable or disable cookies in Firefox 3.6

1. Open the Tools menu and select the Options command.
2. Click on the Privacy tab.
3. Use the "Accept cookies from sites" check box to enable or disable cookies.

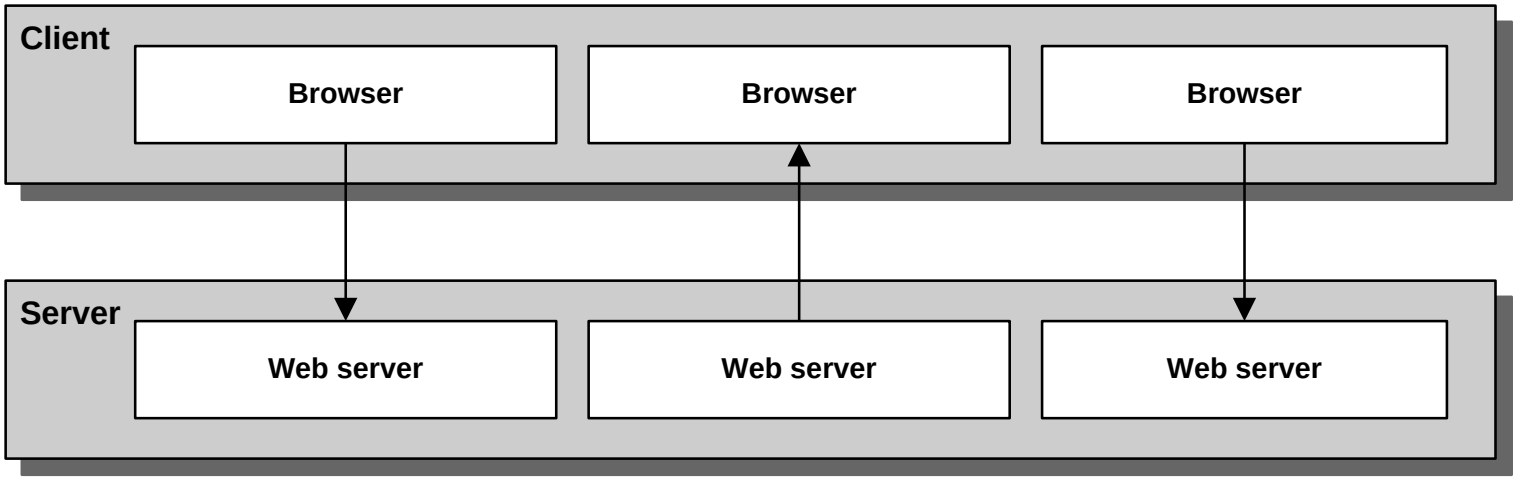
How to enable or disable cookies in IE 8

1. Open the Tools menu and select the Internet Options command.
2. Click the Privacy tab.
3. Use the slider control to enable or disable cookies. To disable cookies, set the security level to "Block All Cookies". To enable cookies, click the Default button to return to default privacy settings.

How to reset default security settings in IE 8

1. Open the Tools menu and select the Internet Options command.
2. Click the Security tab.
3. If not disabled, click the "Reset all zones to default level" button.

Why session tracking is difficult with HTTP

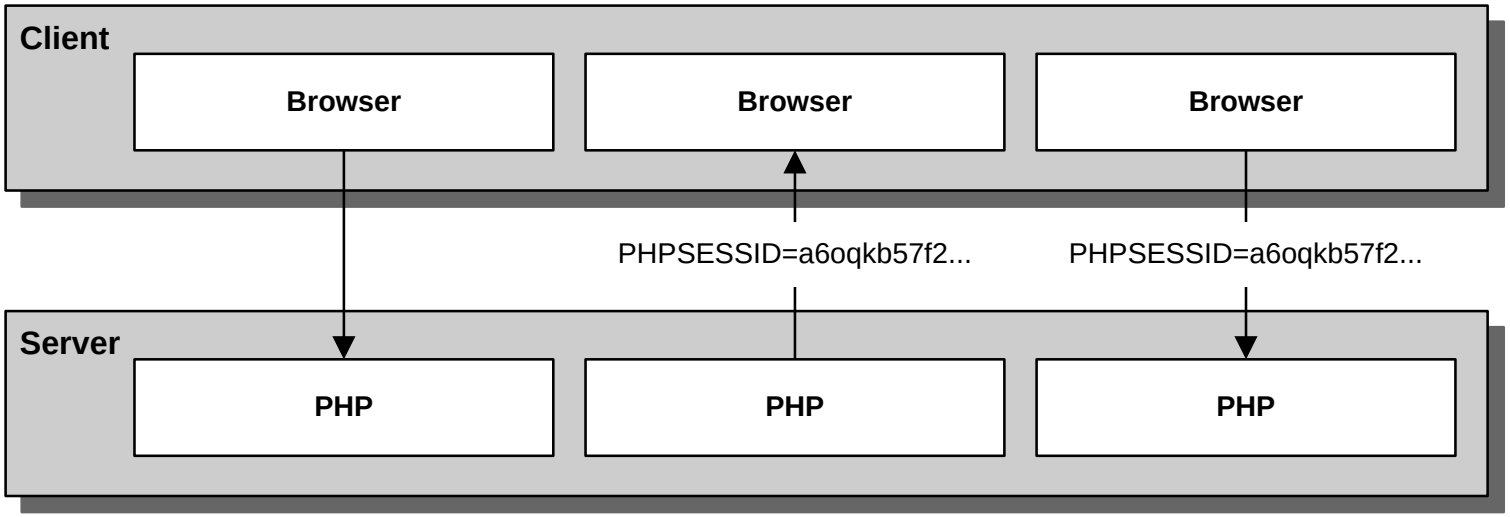


First HTTP Request:
The browser requests a page.

First HTTP Response:
The server returns the requested page and drops the connection.

Following HTTP Requests:
The browser requests a page. The web server has no way to associate the browser with its previous request.

How PHP keeps track of sessions



First HTTP Request:
The browser requests a PHP page. PHP creates a session and assigns it an ID.

First HTTP Response:
The server returns the requested page and the ID for the session as a cookie.

Following HTTP Requests:
The browser requests a PHP page and sends the session ID cookie. PHP uses the session ID to associate the browser with its session.

Start a new session or resume a previous session with the default cookie parameters:

```
session_start();
```

If the session is new, a session ID and session cookie will be created.

To control the session cookie, use the function:

```
session_set_cookie_params($lifetime, $path, $domain,
                          $secure, $httponly)
```

- **$lifetime: of the cookie in seconds; required parameter**
- **$path: the sever path that the cookie is available to; default is current directory of the script setting the cookie.**
- **The other three parameters don't usually need to be changed.**

Start a session with custom cookie parameters:

```
$lifetime = 60 * 60 * 24 * 365;   // 1 year in seconds

session_set_cookie_params($lifetime, '/');

session_start();
```

Note: this must occur before any HTML code is returned and session_set_cookie_params() must precede session_start();

The global $_SESSION variable: an associative array that stores the data for the session.

How to set and get scalar variables

- Set a variable in a session

```
$_SESSION['product_code'] = 'MBT-1753';
```

- Get a variable from a session

```
$product_code = $_SESSION['product_code'];
```

How to set and get arrays

Set an array in a session

```
if (!isset($_SESSION['cart'])) {
    $_SESSION['cart'] = array();
}
```

Add an element to an array that's stored in a session

```
$_SESSION['cart']['key1'] = 'value1';
$_SESSION['cart']['key2'] = 'value2';
```

Get and use an array that's stored in a session

```
$cart = $_SESSION['cart'];
foreach ($cart as $item) {
    echo '<li>' . $item . '</li>';
}
```

How to remove variables from a session

- **Remove a session variable**

```
unset($_SESSION['cart']);
```

- **Remove all session variables**

```
$_SESSION = array();
```

- Note: don't use unset on the entire $_SESSION array, as it causes unpredictable results

- **As the application runs the $_SESSION array is stored in memory**
- **When it ends PHP saves the contents of the $_SESSION array in a file on the web server.**
- **PHP deleted the session data when the session expires which by default is after 24 minutes of inactivity.**
- **In general it is safe to store strings, numbers, Bools, and arrays.**
- **Storing objects requires additional manipulation and will be discussed in Chapter 14.**

How to end a session

```
$_SESSION = array();  //clear session data from memory
session_destroy(); //clean up the session ID
```

However: this doesn't delete the session cookie from the user's browser.

How to completely remove the session data from both the client and the server:

```
Delete the session cookie from the browser
$name = session_name(); // Get name of session cookie

// Create expire date in past
$expire = strtotime('-1 year');

// Get session params
$params = session_get_cookie_params();
$path = $params['path'];
$domain = $params['domain'];
$secure = $params['secure'];
$httponly = $params['httponly'];

setcookie($name, '', $expire, $path, $domain,
        $secure, $httponly);
```

Functions to manage sessions

```
session_name()
session_id([$id])
session_write_close()
session_regenerate_id()
```

Get the name of the session cookie

```
$name = session_name();     // By default, PHPSESSID
```

Get the value of the session ID

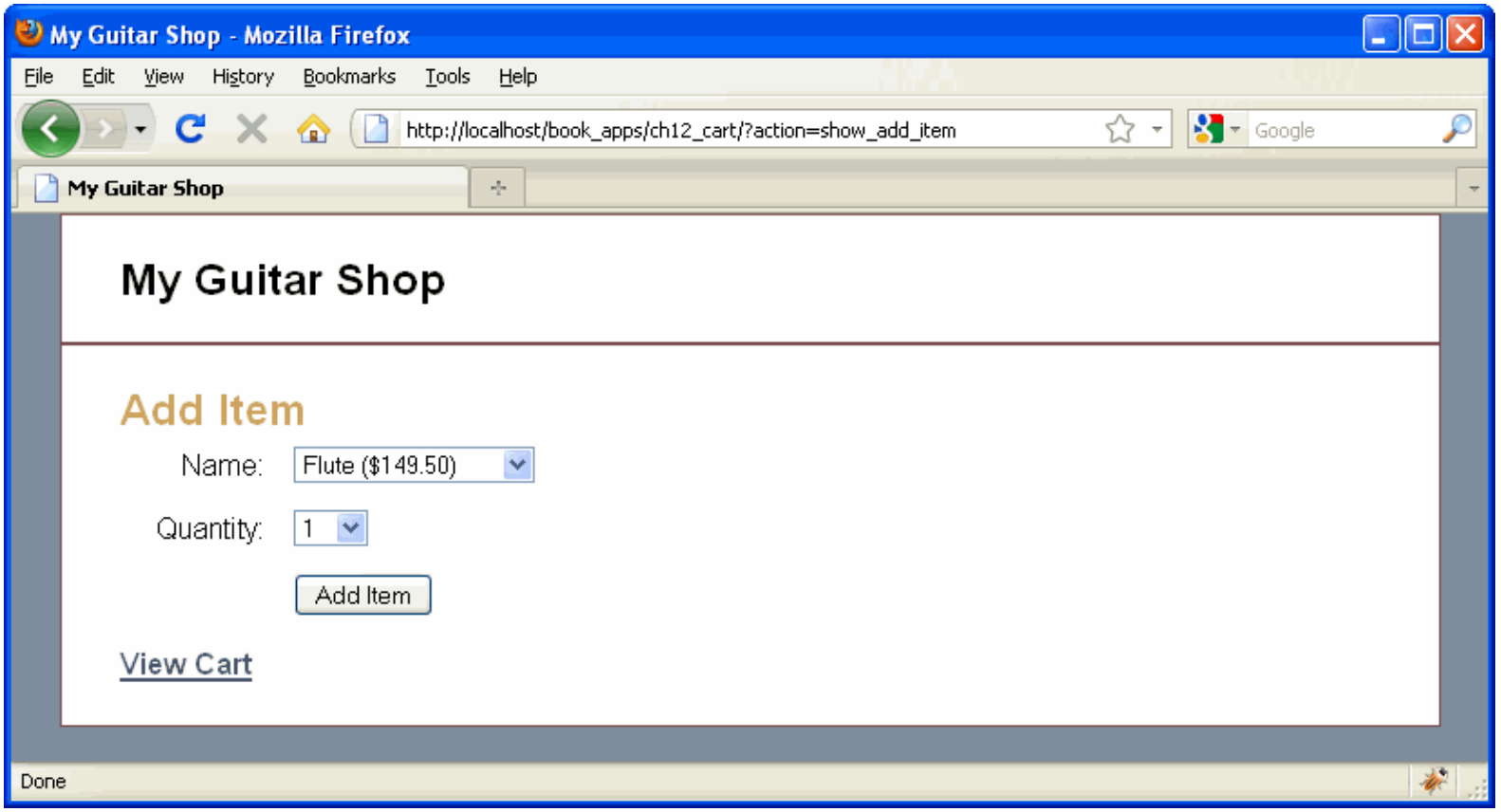
```
$id = session_id();
```

Set the session ID

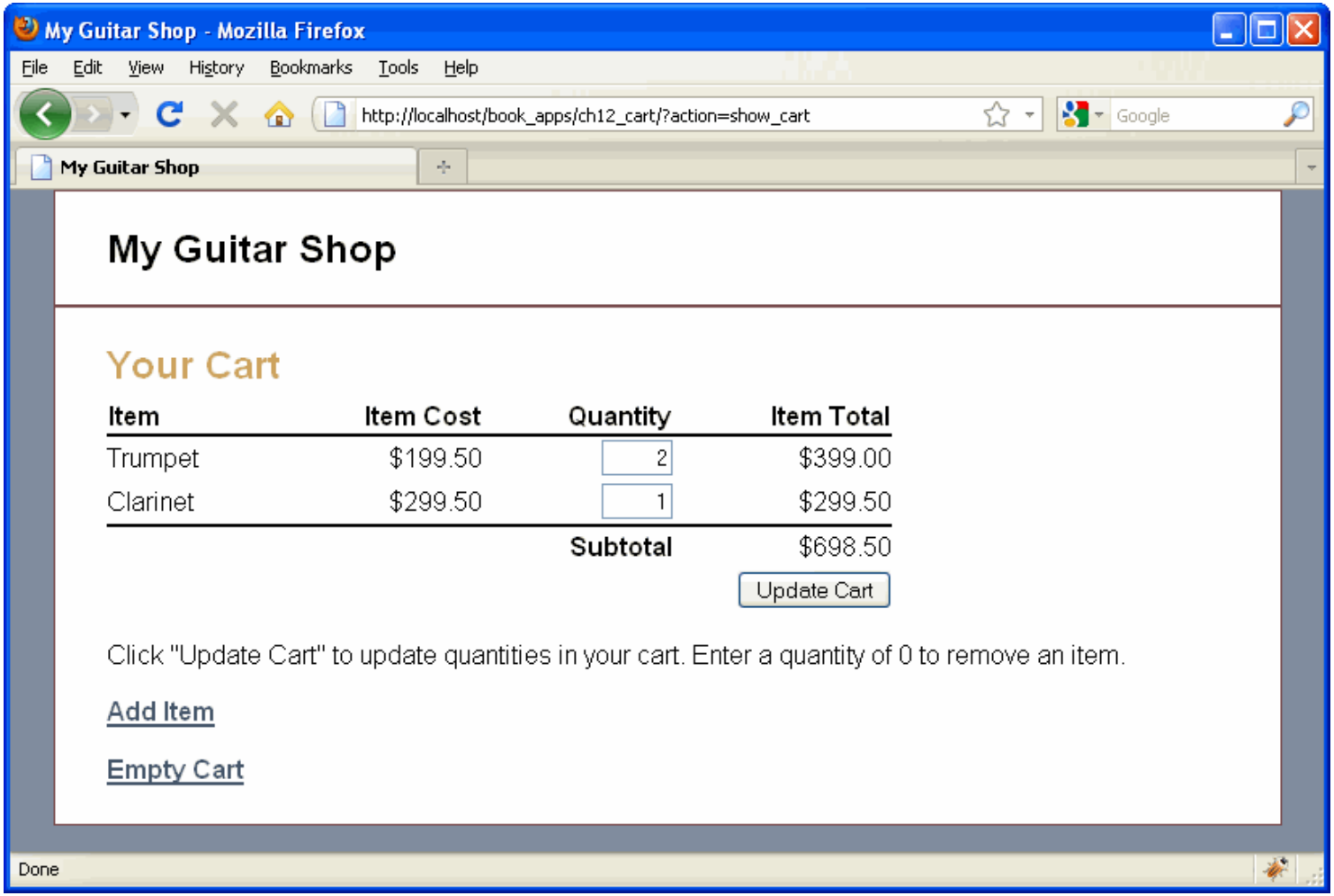
```
session_id('abc123');
```

A Simple Shopping Cart Application

The Add Item page



The Cart page



The index.php file

```
<?php
// Start session management with a persistent cookie
$lifetime = 60 * 60 * 24 * 14;    // 2 weeks in seconds
session_set_cookie_params($lifetime, '/');
session_start();

// Create a cart array if needed
if (empty($_SESSION['cart12']))
    $_SESSION['cart12'] = array();

// Create a table of products
$products = array();
$products['MMS-1754'] =
    array('name' => 'Flute', 'cost' => '149.50');
$products['MMS-6289'] =
    array('name' => 'Trumpet', 'cost' => '199.50');
$products['MMS-3408'] =
    array('name' => 'Clarinet', 'cost' => '299.50');
```

The index.php file (continued)

```
// Include cart functions
require_once('cart.php');

// Get the action to perform
if (isset($_POST['action'])) {
    $action = $_POST['action'];
} else if (isset($_GET['action'])) {
    $action = $_GET['action'];
} else {
    $action = 'show_add_item';
}

// Add or update cart as needed
switch($action) {
    case 'add':
        add_item($_POST['productkey'], $_POST['itemqty']);
        include('cart_view.php');
        break;
```

The index.php file (continued)

```
    case 'update':
        $new_qty_list = $_POST['newqty'];
        foreach($new_qty_list as $key => $qty) {
            if ($_SESSION['cart12'][$key]['qty'] != $qty) {
                update_item($key, $qty);
            }
        }
        include('cart_view.php');
        break;
    case 'show_cart':
        include('cart_view.php');
        break;
    case 'show_add_item':
        include('add_item_view.php');
        break;
    case 'empty_cart':
        unset($_SE['cart12']);
        include('cart_view.php');
        break;
}
?>
```

The cart.php file

```
<?php
// Add an item to the cart
function add_item($key, $quantity) {
    global $products;
    if ($quantity < 1) return;

    // If item already exists in cart, update quantity
    if (isset($_SESSION['cart12'][$key])) {
        $quantity += $_SESSION['cart12'][$key]['qty'];
        update_item($key, $quantity);
        return;
    }
```

The cart.php file (continued)

```
    // Add item
    $cost = $products[$key]['cost'];
    $total = $cost * $quantity;
    $item = array(
        'name' => $products[$key]['name'],
        'cost' => $cost,
        'qty'  => $quantity,
        'total' => $total
    );
    $_SESSION['cart12'][$key] = $item;
}
```

The cart.php file (continued)

```
// Update an item in the cart
function update_item($key, $quantity) {
    global $products;
    $quantity = (int) $quantity;
    if (isset($_SESSION['cart12'][$key])) {
        if ($quantity <= 0) {
            unset($_SESSION['cart12'][$key]);
        } else {
            $_SESSION['cart12'][$key]['qty'] = $quantity;
            $total = $_SESSION['cart12'][$key]['cost'] *
                     $_SESSION['cart12'][$key]['qty'];
            $_SESSION['cart12'][$key]['total'] = $total;
        }
    }
}
```

The cart.php file (continued)

```
// Get cart subtotal
function get_subtotal () {
    $subtotal = 0;
    foreach ($_SESSION['cart12'] as $item) {
        $subtotal += $item['total'];
    }
    $subtotal = number_format($subtotal, 2);
    return $subtotal;
}
?>
```

The add_item_view.php file

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 ...>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title>My Guitar Shop</title>
    <link rel="stylesheet" type="text/css"
          href="main.css"/>
</head>
<body>
    <div id="page">
        <div id="header">
            <h1>My Guitar Shop</h1>
        </div>
        <div id="main">
```

The add_item_view.php file (continued)

```
<h1>Add Item</h1>
<form action="." method="post">
    <input type="hidden" name="action"
           value="add"/>

    <label>Name:</label>
    <select name="productkey">
    <?php foreach($products as $key => $product) :
        $cost = number_format($product['cost'], 2);
        $name = $product['name'];
        $item = $name . ' ($' . $cost . ')';
    ?>
        <option value="<?php echo $key; ?>">
            <?php echo $item; ?>
        </option>
    <?php endforeach; ?>
    </select><br />
```

The add_item_view.php file (continued)

```
                <label>Quantity:</label>
                <select name="itemqty">
                <?php for($i = 1; $i <= 10; $i++) : ?>
                    <option value="<?php echo $i; ?>">
                        <?php echo $i; ?>
                    </option>
                <?php endfor; ?>
                </select><br />

                <label> </label>
                <input type="submit" value="Add Item"/>
            </form>
            <p><a href=".?action=show_cart">
               View Cart</a></p>

        </div><!-- end main -->
    </div><!-- end page -->
</body>
</html>
```

The cart_view.php file

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 ...>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
    <title>My Guitar Shop</title>
    <link rel="stylesheet" type="text/css"
          href="main.css"/>
</head>
<body>
    <div id="page">
        <div id="header">
            <h1>My Guitar Shop</h1>
        </div>
        <div id="main">
```

The cart_view.php file (continued)

```
<h1>Your Cart</h1>
<?php if (count($_SESSION['cart12']) == 0) : ?>
    <p>There are no items in your cart.</p>
<?php else: ?>
    <form action="." method="post">
    <input type="hidden" name="action"
           value="update"/>
    <table>
        <tr id="cart_header">
            <th class="left">Item</th>
            <th class="right">Item Cost</th>
            <th class="right">Quantity</th>
            <th class="right">Item Total</th>
        </tr>
```

The cart_view.php file (continued)

```
<?php foreach( $_SESSION['cart12']
        as $key => $item ) :
    $cost  = number_format($item['cost'],  2);
    $total = number_format($item['total'], 2);
?>
    <tr>
        <td>
            <?php echo $item['name']; ?>
        </td>
        <td class="right">
            $<?php echo $cost; ?>
        </td>
        <td class="right">
           <input type="text"
              class="cart_qty"
              name=
                "newqty[<?php echo $key; ?>]"
              value=
                "<?php echo $item['qty']; ?>"/>
        </td>
```

The cart_view.php file (continued)

```
                <td class="right">
                        $<?php echo $total; ?>
                    </td>
                </tr>
            <?php endforeach; ?>
                <tr id="cart_footer">
                    <td colspan="3"><b>Subtotal</b></td>
                <td>$<?php echo get_subtotal(); ?></td>
                </tr>
                <tr>
                    <td colspan="4" class="right">
                        <input type="submit"
                               value="Update Cart"/>
                    </td>
                </tr>
            </table>
            <p>Click "Update Cart" to update quantities.
               Enter a quantity of 0 to remove an item.
            </p>
            </form>
        <?php endif; ?>
```

The cart_view.php file (continued)

```
            <p><a href=".?action=show_add_item">
                  Add Item</a></p>
            <p><a href=".?action=empty_cart">
                  Empty Cart</a></p>

        </div><!-- end main -->
    </div><!-- end page -->
</body>
</html>
```