**index.php**

<?php

// Start session management

session\_start();

// Create a cart array if needed

if (empty($\_SESSION['cart13'])) { $cart = array(); }

else

$cart = $\_SESSION['cart13'];

// 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');

// Include cart functions

require\_once('cart.php');

// Get the sort key

$sort\_key = filter\_input(INPUT\_POST, 'sortkey');

if ($sort\_key === NULL) { $sort\_key = 'name'; }

// Get the action to perform

$action = filter\_input(INPUT\_POST, 'action');

if ($action === NULL) {

$action = filter\_input(INPUT\_GET, 'action');

if ($action === NULL) {

$action = 'show\_add\_item';

}

}

// Add or update cart as needed

switch($action) {

case 'add':

$key = filter\_input(INPUT\_POST, 'productkey');

$quantity = filter\_input(INPUT\_POST, 'itemqty');

cart\add\_item($cart, $key, $quantity);

include('cart\_view.php');

break;

case 'update':

$new\_qty\_list = filter\_input(INPUT\_POST, 'newqty',

FILTER\_DEFAULT, FILTER\_REQUIRE\_ARRAY);

foreach($new\_qty\_list as $key => $qty) {

if ($cart[$key]['qty'] != $qty) {

cart\update\_item($cart, $key, $qty);

}

}

cart\sort($cart, $sort\_key);

include('cart\_view.php');

break;

case 'show\_cart':

cart\sort($cart, $sort\_key);

include('cart\_view.php');

break;

case 'show\_add\_item':

include('add\_item\_view.php');

break;

case 'empty\_cart':

unset($cart);

include('cart\_view.php');

break;

}

if (isset($cart))

$\_SESSION['cart13'] = $cart;

else

unset($\_SESSION['cart13']);

?>

**cart.php**

<?php

namespace cart {

// Add an item to the cart

function add\_item(&$cart, $key, $quantity) {

global $products;

if ($quantity < 1) return;

// If item already exists in cart, update quantity

if (isset($cart[$key])) {

$quantity += $cart[$key]['qty'];

update\_item($cart, $key, $quantity);

return;

}

// Add item

$cost = $products[$key]['cost'];

$total = $cost \* $quantity;

$item = array(

'name' => $products[$key]['name'],

'cost' => $cost,

'qty' => $quantity,

'total' => $total

);

$cart[$key] = $item;

}

// Update an item in the cart

function update\_item(&$cart, $key, $quantity) {

$quantity = (int) $quantity;

if (isset($cart[$key])) {

if ($quantity <= 0) {

unset($cart[$key]);

} else {

$cart[$key]['qty'] = $quantity;

$total = $cart[$key]['cost'] \*

$cart[$key]['qty'];

$cart[$key]['total'] = $total;

}

}

}

// Get cart subtotal

function get\_subtotal($cart) {

$subtotal = 0;

foreach ($cart as $item) {

$subtotal += $item['total'];

}

$subtotal\_f = number\_format($subtotal, 2);

return $subtotal\_f;

}

// Get a function for sorting the cart on the specified key

function compare\_factory($sort\_key) {

return function($left, $right) use ($sort\_key) {

if ($left[$sort\_key] == $right[$sort\_key]) {

return 0;

} else if ($left[$sort\_key] < $right[$sort\_key]) {

return -1;

} else {

return 1;

}

};

}

// Sort the cart on the specified key

function sort(&$cart, $sort\_key) {

$compare\_function = compare\_factory($sort\_key);

usort($cart, $compare\_function);

}

}

?>