Chapter 4

How to use PHP with a MySQL database

Objectives

Applied

1. Given the specifications for a database application that requires only the skills that are presented in this chapter, develop the application. That includes:
    - Connecting to a MySQL database
    - Executing SELECT, INSERT, UPDATE, and DELETE statements
    - Handling PDO exceptions
    - Getting the data from the result sets that are returned by SQL statements

Objectives (continued)

Knowledge

1. Describe the PHP code for creating a PDO object that connects to a MySQL database.
2. Describe the use of the PDO methods for executing a SELECT, INSERT, UPDATE, or DELETE statement.
3. Describe the PHP code for handling the PDO exceptions that may occur when you try to create a PDO object.
4. Describe a PHP array and the way that numeric and string indexes are used to access the data in a PHP array.
5. Describe the way the fetch method of a PDO statement object is used to get data from the first row of a result set and the way a foreach statement is used to get the data from all the rows of a result set.

The syntax for creating an object from any class

```
new ClassName(arguments);
```

The syntax for creating a database object from the PDO (PHP Data Objects) class

```
new PDO($dsn, $username, $password);
```

The syntax for a DSN (Data Source Name) for a MySQL database

```
mysql:host=host_address;dbname=database_name
```

How to connect to a MySQL database

```
$dsn = 'mysql:host=localhost;dbname=abc1234';
$username = 'abc1234'; //your username
$password = 'xxxxxx'; //your sql password

// creates PDO object
$db = new PDO($dsn, $username, $password);
```

The syntax for executing a method of any object

```
$objectName->methodName(argumentList)
```

A method of the PDO class for executing a SELECT statement

```
query($select_statement)
```

The syntax for executing the query method of the database object

```
$PDO_object->query($select_statement)
```

A query method with the SELECT statement in a variable

```
$query = 'SELECT * FROM products
          WHERE categoryID = 1
          ORDER BY productID';
$products = $db->query($query);
```

A query method with the SELECT statement as the argument

```
$products = $db->query('SELECT * FROM products');
```

A method of the PDO class for modifying the database

```
exec($sql_statement)//returns the number of rows affected
```

How to execute an INSERT statement

```
$category_id = 1;
$code = 'strat';
$name = 'Fender Stratocaster';
$price = 699.99;

$query = "INSERT INTO products
     (categoryID, productCode, productName, listPrice)
          VALUES
     ($category_id, '$code', '$name', $price)";

$insert_count = $db->exec($query);
```

How to execute an UPDATE statement

```
$product_id = 4;
$price = 599.99;

$query = "UPDATE products
          SET listPrice = $price
          WHERE productID = $product_id";

$update_count = $db->exec($query);
```

How to execute a DELETE statement

```
$product_id = 4;

$query = "DELETE FROM products
          WHERE productID = $product_id";

$delete_count = $db->exec($query);
```

How to display the row counts

```
<p>Insert count: <?php echo $insert_count; ?></p>
<p>Update count: <?php echo $update_count; ?></p>
<p>Delete count: <?php echo $delete_count; ?></p>
```

When a PDO object can’t be created, the class throws an exception. To handle exceptions, use a try/catch statement:

```
try {
    // statements that might throw an exception
} catch (ExceptionClass $exception_name) {
    // statements that handle the exception
}
```

How to handle a PDO exception

```
try {
    $db = new PDO($dsn, $username, $password);
    echo '<p>You are connected to the database!</p>';
} catch (PDOException $e) {
    $error_message = $e->getMessage();
    echo "<p>An error occurred while connecting to
             the database: $error_message </p>";
}
```

A method of the PDOStatement class for getting an array for a row

```
fetch() /*returns an array for the next row in the result
set indexed by column name as a string or by column
position as a numeric index. FALSE is returned of no
array is available. /*
```

Code that gets a result set containing one row

```
$query = 'SELECT productCode, productName, listPrice
          FROM products
          WHERE productID = $productID';

$products = $db->query($query);
// $products is a PDOStatement object

$product = $products->fetch();
// $product is the array for the first row
```

Code that uses a string index to get each column

```
$product_code = $product['productCode'];
$product_name = $product['productName'];
$product_list_price = $product['listPrice'];
```

Code that uses a numeric index to get each column

```
$product_code = $product[0];
$product_name = $product[1];
$product_list_price = $product[2];
```

A query method that returns a result set of two or more rows

```
$query = 'SELECT productCode, productName, listPrice
          FROM products
          WHERE categoryID = 1;'

$products = $db->query($query);
// $products contains the result set
```

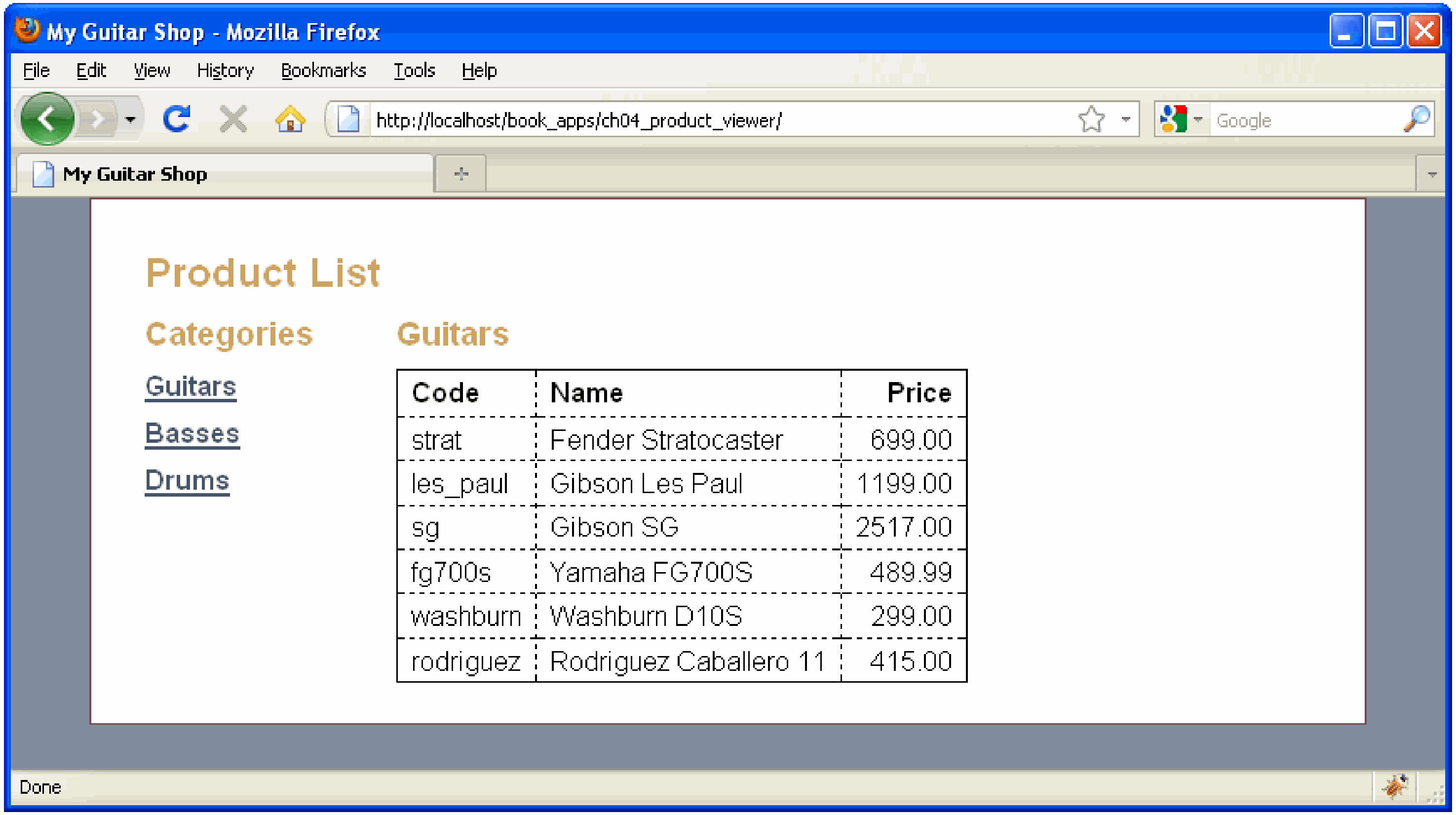
How to use a foreach statement to display the result set in an HTML table

```
<?php foreach ($products as $product) { ?>
<tr>
    <td><?php echo $product['productCode']; ?></td>
    <td><?php echo $product['productName']; ?></td>
    <td><?php echo $product['listPrice']; ?></td>
</tr>
<?php } ?>
```

Another syntax for the foreach statement that works better within PHP tags

```
<?php foreach ($products as $product) : ?>
<tr>
    <td><?php echo $product['productCode']; ?></td>
    <td><?php echo $product['productName']; ?></td>
    <td><?php echo $product['listPrice']; ?></td>
</tr>
<?php endforeach; ?>
```

The user interface



My Guitar Shop - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost/book_apps/ch04_product_viewer/
Google
My Guitar Shop
Done

Product List

Categories

Guitars

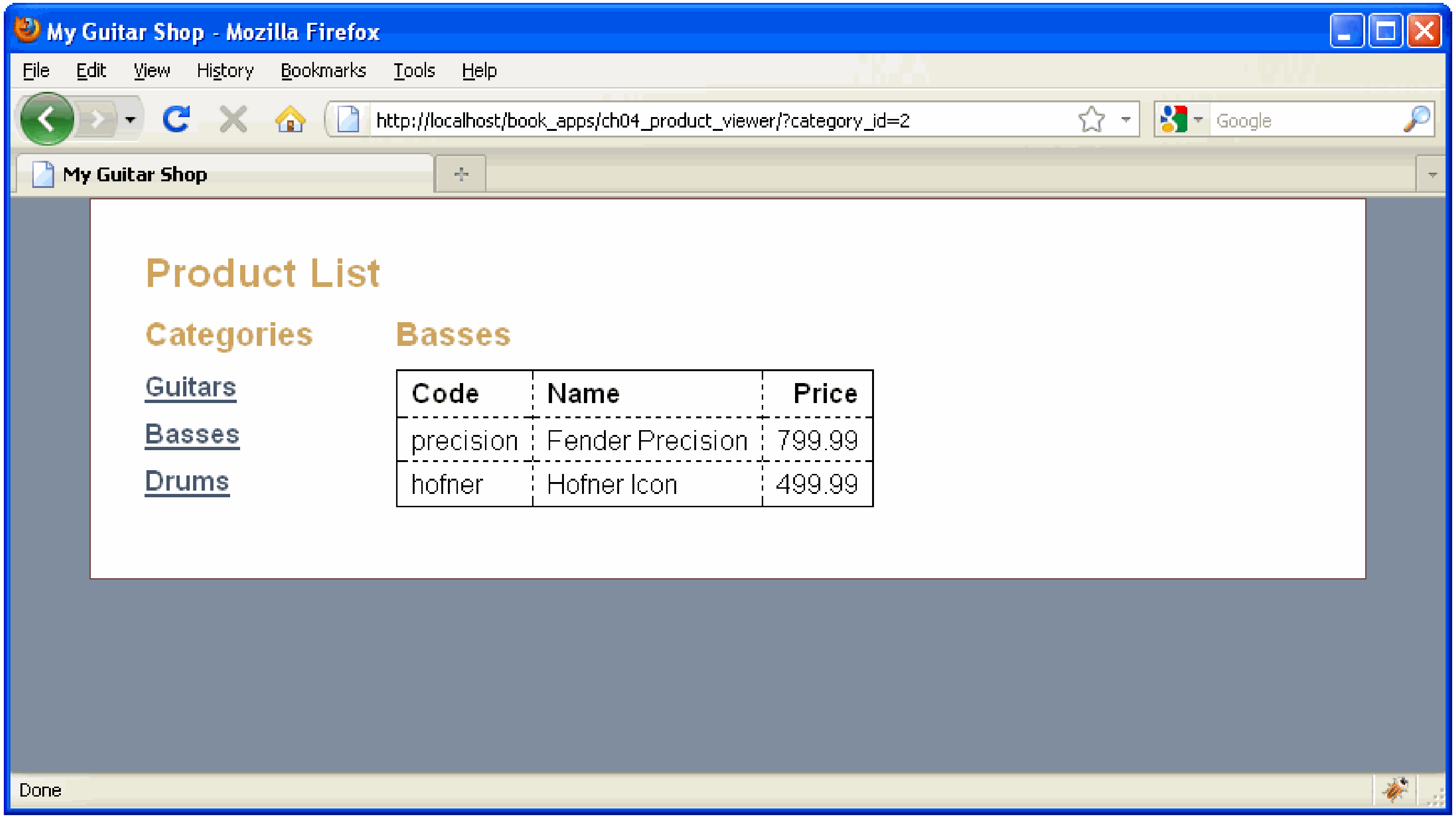
Basses

Drums

Guitars

Code	Name	Price
strat	Fender Stratocaster	699.00
les_paul	Gibson Les Paul	1199.00
sg	Gibson SG	2517.00
fg700s	Yamaha FG700S	489.99
washburn	Washburn D10S	299.00
rodriguez	Rodriguez Caballero 11	415.00

The user interface after the user selects a new category


My Guitar Shop - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost/book_apps/ch04_product_viewer/?category_id=2
Google
My Guitar Shop
Done

Product List

Categories

- Guitars
- Basses
- Drums

Basses

Code	Name	Price
precision	Fender Precision	799.99
hofner	Hofner Icon	499.99

The database.php file

```
<?php
    $dsn = 'mysql:host=localhost;dbname=my_guitar_shop1';
    $username = 'mgs_user';
    $password = 'pa55word';

    try {
        $db = new PDO($dsn, $username, $password);
    } catch (PDOException $e) {
        $error_message = $e->getMessage();
        include('database_error.php');
        exit();
    }
?>
```

The database_error.php file

```
<!-- the head section -->
<head>
    <title>My Guitar Shop</title>
    <link rel="stylesheet" type="text/css"
          href="main.css" />
</head>

<!-- the body section -->
<body>
<div id="page">
    <div id="main">
        <h1>Database Error</h1>
        <p>There was a database connection error.</p>
        <p>The database must be installed.</p>
        <p>MySQL must be running.</p>
        <p>Error message:
            <?php echo $error_message; ?></p>
    </div>
</div><!-- end page -->
</body>
```

The index.php file

```
<?php
    require 'database.php';

    // Get category ID
    $category_id = $_GET['category_id'];
    if (!isset($category_id)) {
        $category_id = 1;
    }

    // Get name for current category
    $query = "SELECT * FROM categories
              WHERE categoryID = $category_id";
    $category = $db->query($query);
    $category = $category->fetch();
    $category_name = $category['categoryName'];

    // Get all categories
    $query = 'SELECT * FROM categories
              ORDER BY categoryID';
    $categories = $db->query($query);
```

The index.php file (continued)

```
    // Get products for selected category
    $query = "SELECT * FROM products
              WHERE categoryID = $category_id
              ORDER BY productID";
    $products = $db->query($query);
?>
```

The index.php file (continued)

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

    <!-- the body section -->
    <body>
    <div id="page">
        <div id="main">

        <h1>Product List</h1>
```

The index.php file (continued)

```
<div id="sidebar">
    <!-- display a list of categories -->
    <h2>Categories</h2>
    <ul class="nav">
    <?php foreach ($categories as $category) : ?>
        <li>
        <a href="?category_id=
            <?php echo $category['categoryID']; ?>">
            <?php echo $category['categoryName']; ?>
        </a>
        </li>
    <?php endforeach; ?>
    </ul>
</div>
```

The index.php file (continued)

```
<div id="content">
    <!-- display a table of products -->
    <h2><?php echo $category_name; ?></h2>
    <table>
        <tr>
            <th>Code</th>
            <th>Name</th>
            <th class="right">Price</th>
        </tr>
        <?php foreach ($products as $product) : ?>
        <tr>
           <td><?php echo
               $product['productCode']; ?></td>
           <td><?php echo
               $product['productName']; ?></td>
           <td class="right"><?php echo
               $product['listPrice']; ?></td>
        </tr>
        <?php endforeach; ?>
    </table>
</div>
```

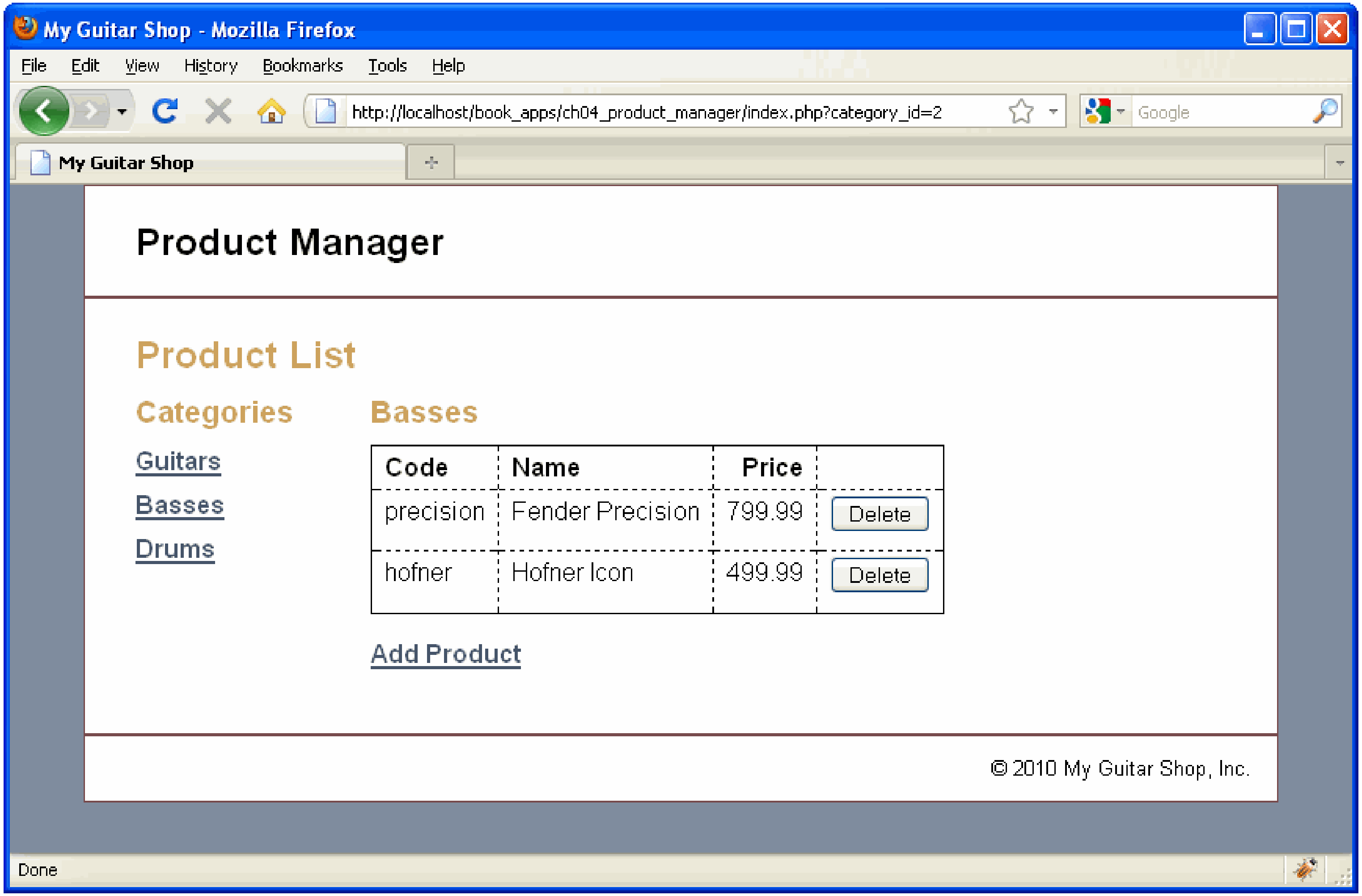
The index.php file (continued)

```
        </div><!-- end main -->

        <div id="footer"></div>

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

The Product List page



My Guitar Shop - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://localhost/book_apps/ch04_product_manager/index.php?category_id=2
Google
My Guitar Shop
Done

Product Manager

Product List

Categories

- Guitars
- Basses
- Drums

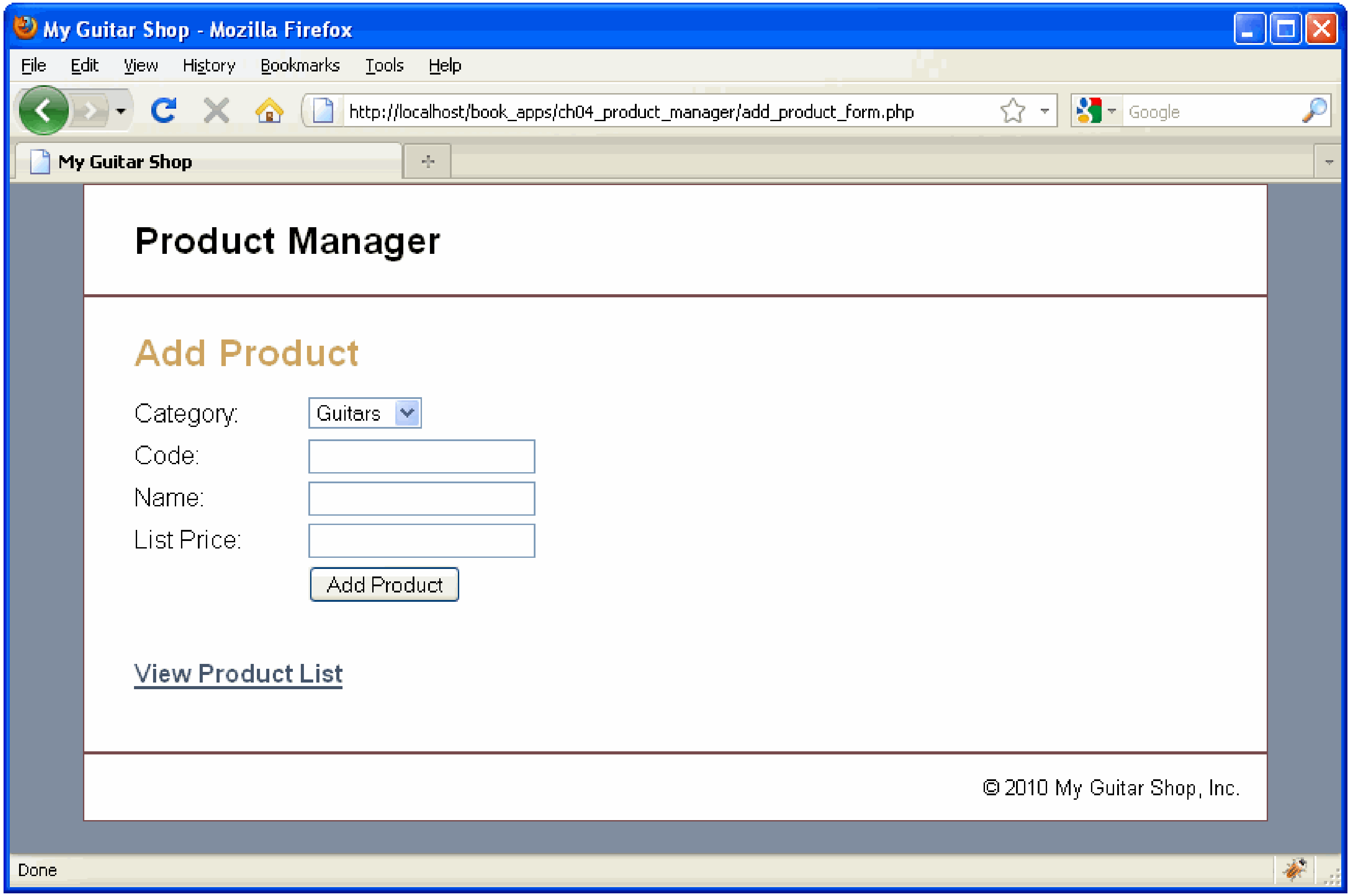
Basses

Code	Name	Price	
precision	Fender Precision	799.99	Delete
hofner	Hofner Icon	499.99	Delete

Add Product

© 2010 My Guitar Shop, Inc.

The Add Product page



My Guitar Shop - Mozilla Firefox

http://localhost/book_apps/ch04_product_manager/add_product_form.php

Product Manager

Add Product

Category: Guitars

Code:

Name:

List Price:

The index.php file

```
<?php
    require_once('database.php');

    // Get category ID
    if(!isset($category_id)) {
        $category_id = $_GET['category_id'];
        if (!isset($category_id)) {
            $category_id = 1;
        }
    }

    // Get name for current category
    $query = "SELECT * FROM categories
              WHERE categoryID = $category_id";
    $category = $db->query($query);
    $category = $category->fetch();
    $category_name = $category['categoryName'];
```

The index.php file (continued)

```
    // Get all categories
    $query = 'SELECT * FROM categories
              ORDER BY categoryID';
    $categories = $db->query($query);

    // Get products for selected category
    $query = "SELECT * FROM products
              WHERE categoryID = $category_id
              ORDER BY productID";
    $products = $db->query($query);
?>
```

The index.php file (continued)

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 ...>
<html xmlns="http://www.w3.org/1999/xhtml">

<!-- the head section -->
<head>
    <title>My Guitar Shop</title>
    <link rel="stylesheet" type="text/css"
          href="main.css" />
</head>
```

The index.php file (continued)

```
<body>
    <div id="page">
    <div id="header">
        <h1>Product Manager</h1>
    </div>
    <div id="main">
        <h1>Product List</h1>
        <div id="sidebar">
            <!-- display a drop-down list of categories -->
            <h2>Categories</h2>
            <ul class="nav">
            <?php foreach ($categories as $category) : ?>
                <li>
                <a href="?category_id=
                    <?php echo $category['categoryID']; ?>">
                    <?php echo $category['categoryName']; ?>
                </a>
                </li>
            <?php endforeach; ?>
            </ul>
        </div>
```

The index.php file (continued)

```
<div id="content">
    <!-- display a table of products -->
    <h2><?php echo $category_name; ?></h2>
    <table>
        <tr>
            <th>Code</th>
            <th>Name</th>
            <th class="right">Price</th>
            <th>&nbsp;</th>
        </tr>
        <?php foreach ($products as $product) : ?>
        <tr>
           <td><?php echo
                $product['productCode']; ?></td>
           <td><?php echo $product['productName'];
                ?></td>
           <td class="right"><?php echo
                $product['listPrice']; ?></td>
```

The index.php file (continued)

```
               <td><form action="delete_product.php"
                         method="post"
                         id="delete_product_form">
                   <input type="hidden"
                       name="product_id"
                       value="<?php echo
                         $product['productID']; ?>" />
                   <input type="hidden"
                       name="category_id"
                       value="<?php echo
                         $product['categoryID']; ?>" />
                   <input type="submit"
                       value="Delete" />
                </form></td>
            </tr>
            <?php endforeach; ?>
        </table>
        <p><a href="add_product_form.php">
              Add Product</a></p>
    </div>
</div>
```

The index.php file (continued)

```
    <div id="footer">
        <p>&copy; <?php echo date("Y"); ?>
            My Guitar Shop, Inc.</p>
    </div>

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

The delete_product.php file

```
<?php
    // Get IDs
    $product_id = $_POST['product_id'];
    $category_id = $_POST['category_id'];

    // Delete the product from the database
    require_once('database.php');
    $query = "DELETE FROM products
              WHERE productID = '$product_id'";
    $db->exec($query);

    // Display the Product List page
    include('index.php');
?>
```

The add_product_form.php file

```
<?php
    require_once('database.php');
    $query = 'SELECT *
              FROM categories
              ORDER BY categoryID';
    $categories = $db->query($query);
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 ...>
<html xmlns="http://www.w3.org/1999/xhtml">

<!-- the head section -->
<head>
    <title>My Guitar Shop</title>
    <link rel="stylesheet" type="text/css"
          href="main.css" />
</head>
```

The add_product_form.php file (continued)

```
<!-- the body section -->
<body>
    <div id="page">
        <div id="header">
            <h1>Product Manager</h1>
        </div>

        <div id="main">
            <h1>Add Product</h1>
            <form action="add_product.php" method="post"
                  id="add_product_form" >

              <label>Category:</label>
              <select name="category_id">
              <?php foreach ($categories as $category) : ?>
                <option value="<?php echo
                    $category['categoryID']; ?>">
                    <?php echo $category['categoryName']; ?>
                </option>
              <?php endforeach; ?>
              </select><br />
```

The add_product_form.php file (continued)

```
              <label>Code:</label>
              <input type="input" name="code" />
              <br />

              <label>Name:</label>
              <input type="input" name="name" />
              <br />

              <label>List Price:</label>
              <input type="input" name="price" />
              <br />

              <label>&nbsp;</label>
              <input type="submit" value="Add Product" />
              <br />
            </form>
            <p><a href="index.php">View Product List</a></p>
        </div><!-- end main -->
    </div><!-- end page -->
</body>
</html>
```

The add_product.php file

```
<?php
    // Get the product data
    $category_id = $_POST['category_id'];
    $code = $_POST['code'];
    $name = $_POST['name'];
    $price = $_POST['price'];

// Validate inputs
if (empty($code) || empty($name) || empty($price) ) {
    $error = "Invalid product data. Try again.";
    include('error.php');
} else {
    // If valid, add the product to the database
    require_once('database.php');
    $query = "INSERT INTO products
        (categoryID, productCode, productName, listPrice)
              VALUES
        ('$category_id', '$code', '$name', '$price')";
    $db->exec($query);
```

The add_product.php file (continued)

```
    // Display the Product List page
    include('index.php');
}
?>
```