Using MySQL

PHP Data Objects

- PDO (PHP Data Objects) supports most popular databases because it defines a consistent interface for accessing databases, the same PHP code can be used with more than one type of database.
- To create a PDO object that connects to a MySQL database, you use the PDO class with 3 arguments: DSN (Data Source Name), username, and password.
  - Syntax for creating a database object from the PDO class
    - new PDO($dsn, $username, $password);
  - Syntax for a DSN for a MySQL database.
    - mysql:host=host_address;dbname=database_name
  - Connect to a MySQL database named bookDB

```
$dsn = 'mysql:host=localhost;dbname=book_db1';
$username = 'admin';
$password = 'pass@word';
$db = new PDO($den, $username, $password);
```

Executing SQL Statement

- To call a method from any object, you code the name of the object, followed by -> , followed by the name of the method.
- To execute a SELECT statement, use the query method of the PDO object. It takes only one argument that is the SELECT statement to be executed.
- If the statement returns a result set, the query returns the result set in a PDOStatement object.
- To execute an INSERT, UPDATE, or DELETE statement, you use the exec method of the PDO object with the SQL statement as the argument.
    - Each of these methods returns a value that represents the number of rows that were affected, and this value can be assigned to a variable.

Example of the query() method

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset="UTF-8">
      <title>Select Example</title>
   </head>
   <body>
      <?php
      $dsn = 'mysql:host=localhost;dbname=book_db1';
      $username = 'admin';
      $password = 'pass@word';

      try {
         $db = new PDO($dsn, $username, $password);
      } catch (PDOException $ex) {
         $error_msg = $ex->getMessage();
         echo $error_msg;
         exit();
      }

      // Get the current publisher's name
      $query = "SELECT publisherName FROM publishers WHERE publisherID = 1";
      $publisher = $db->query($query);
      $publisher = $publisher->fetch();
      $publiserName = $publisher['publisherName'];
      echo "<p>$publiserName</p>";
      ?>
   </body>
</html>
```

The exec_examples.php

```
<!DOCTYPE html>
<html>
   <head>
      <meta charset="UTF-8">
      <title>Query Example</title>
   </head>
   <body>
      <?php
      $dsn = 'mysql:host=localhost;dbname=book_db1';
      $username = 'admin';
      $password = 'pass@word';

      try {
         $db = new PDO($dsn, $username, $password);
      } catch (PDOException $ex) {
         $error_msg = $ex->getMessage();
         echo $error_msg;
         exit();
      }

      $query = "INSERT INTO books
      (publisherID, isbn, bookTile, bookPrice)
      VALUES
      (2, '888888', 'ASP', '78.9')";
      $insert_count = $db->exec($query);
      echo "<p>$insert_count inserted.</p>";

      $query = "UPDATE books
      SET bookPrice = 100
      WHERE isbn = '888888'" ;
      $upate_count = $db->exec($query);
      echo "<p>$upate_count updated.</p>";

      $query = "DELETE FROM books
      WHERE isbn = '888888'" ;
      $delete_count = $db->exec($query);
      echo "<p>$delete_count deleted.</p>";
      ?>
   </body>
</html>
```

try/catch statement

- To handle exceptions, you use try/catch statement. First, you code try block around any PHP statements that might throw an exception. Then you code a catch block that catches the exception. This is known as exception handling.
- The base class is the Exception class. The PDOException class is used for errors thrown by the PDO library.
- All Exception objects provide a getMessage method that lets you get the error message.

```
try {
    $db = new PDO($dsn, $username, $password);
    echo 'Connected.';
} catch (PDOException $ex) {
    $error_msg = $ex->getMessage();
    include('db_error.php');
    exit();
}
```

The fetch() method

- The fetch method of a PDOStatement object allows you to get an array for the first row or next row of a result set. Then you can use column names or numeric indexes to access the data that‘s stored in that row. If no array is available, this method returns false value.
- Examples

```
$query = "SELECT publisherName FROM publishers
        WHERE publisherID = $publisher_id";
$publisher = $db->query($query);
$publisher = $publisher->fetch();
$publiserName = $publisher['publisherName'];
```

Iterating through a result set

- The result set will be returned in a PDOStatement object when the query method of a database object is executed. Then you can use a foreach statement to get the data from the result set.
- The foreach statement calls the fetch method the PDOStatement object automatically as it iterates through the rows in the result set.

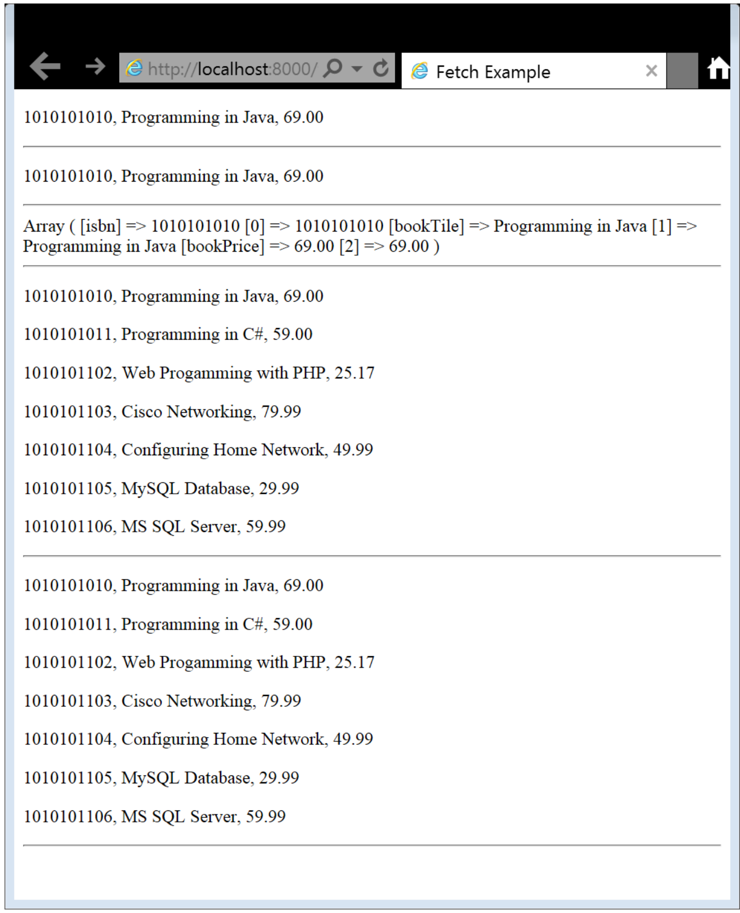
```
foreach ($publishers as $publisher) :
        echo $publisher['publisherID'];
        echo $publisher['publisherName']; ?>
endforeach;
```

OR

```
foreach ($publishers as $publisher) {
        echo $publisher['publisherID'];
        echo $publisher['publisherName']; ?>
}
```

The fetch_example.php

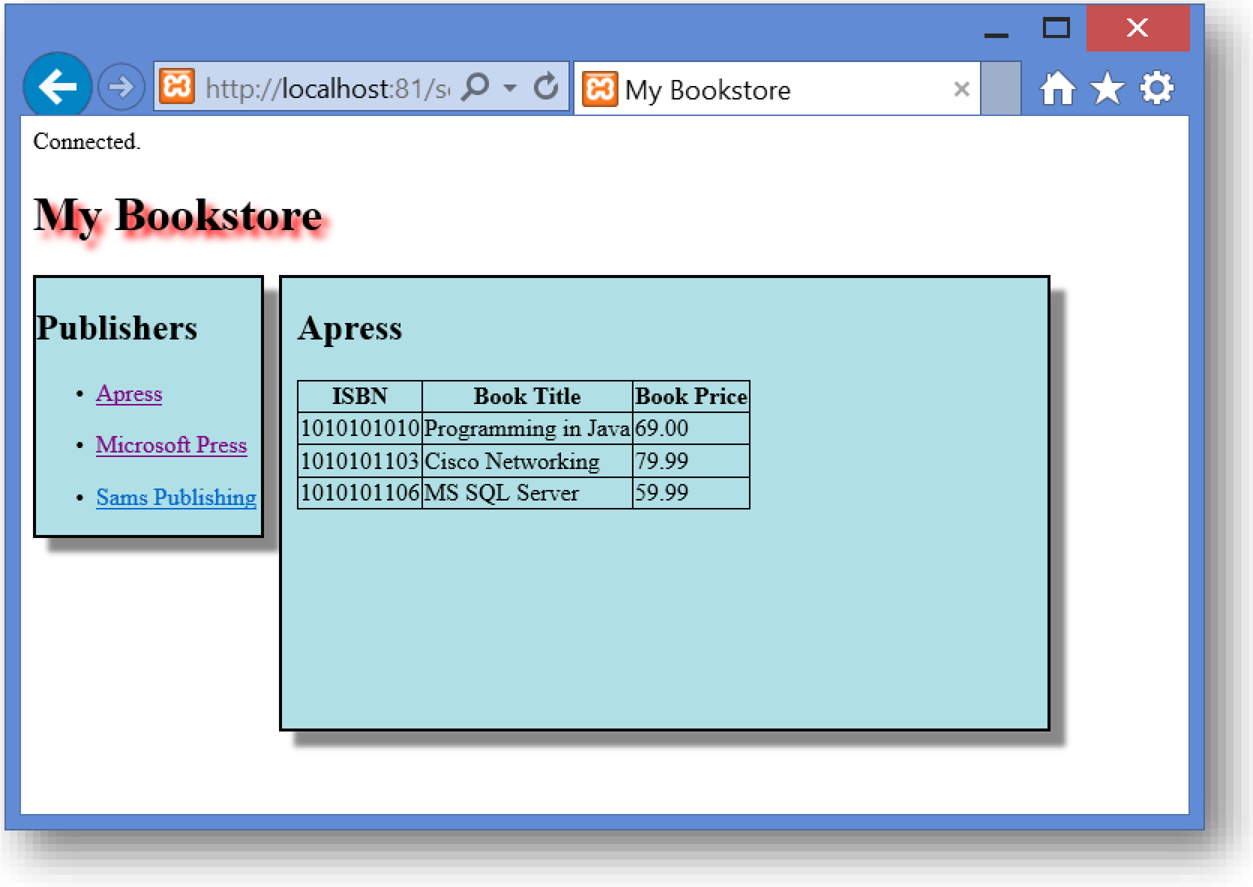
```
<!DOCTYPE html>
<html>
   <head>
      <meta charset="UTF-8">
      <title>Fetch Example</title>
   </head>
   <body>
      <?php
      $dsn = 'mysql:host=localhost;dbname=book_db1';
      $username = 'admin';
      $password = 'pass@word';

      try {
         $db = new PDO($dsn, $username, $password);
      } catch (PDOException $ex) {
         echo $ex->getMessage();
      }

      // Get the current publisher's name
      $query = "SELECT isbn, bookTile, bookPrice FROM books";
      $books = $db->query($query);
      // Get the first row
      $book = $books->fetch();
      // To print the first row
      echo "<p>" . $book['isbn'] . ", " . $book['bookTile'] . ", " . $book['bookPrice'] . "</p>";
      echo '<hr>';
      // Alternativey
      echo "<p>" . $book[0] . ", " . $book[1] . ", " . $book[2] . "</p>";
      echo '<hr>';
      // Another way
      print_r($book);
      echo '<hr>';
      // print all rowa
      $books = $db->query($query);
      foreach ($books as $book) {
         echo "<p>" . $book['isbn'] . ", " . $book['bookTile'] . ", " . $book['bookPrice'] . "</p>";
      }
      echo '<hr>';
      // print all rowa
      $books = $db->query($query);
      $book = $books->fetch();
      while ($book ) {
         echo "<p>" . $book['isbn'] . ", " . $book['bookTile'] . ", " . $book['bookPrice'] . "</p>";
         $book = $books->fetch();
      }
      echo '<hr>';
      ?>
   </body>
</html>
```



```
1010101010, Programming in Java, 69.00
----------------------------------------
1010101010, Programming in Java, 69.00
----------------------------------------
Array ( [isbn] => 1010101010 [0] => 1010101010 [bookTile] => Programming in Java [1] =>
Programming in Java [bookPrice] => 69.00 [2] => 69.00 )
----------------------------------------
1010101010, Programming in Java, 69.00

1010101011, Programming in C#, 59.00

1010101102, Web Progamming with PHP, 25.17

1010101103, Cisco Networking, 79.99

1010101104, Configuring Home Network, 49.99

1010101105, MySQL Database, 29.99

1010101106, MS SQL Server, 59.99
----------------------------------------
1010101010, Programming in Java, 69.00

1010101011, Programming in C#, 59.00

1010101102, Web Progamming with PHP, 25.17

1010101103, Cisco Networking, 79.99

1010101104, Configuring Home Network, 49.99

1010101105, MySQL Database, 29.99

1010101106, MS SQL Server, 59.99
----------------------------------------
```

The Search Book Application

- You can search a book by publisher.



Connected.

My Bookstore

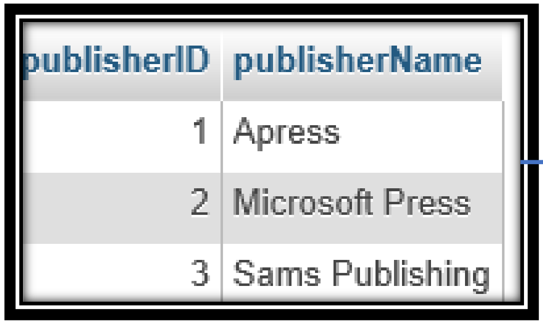
Publishers

- Apress
- Microsoft Press
- Sams Publishing

Apress

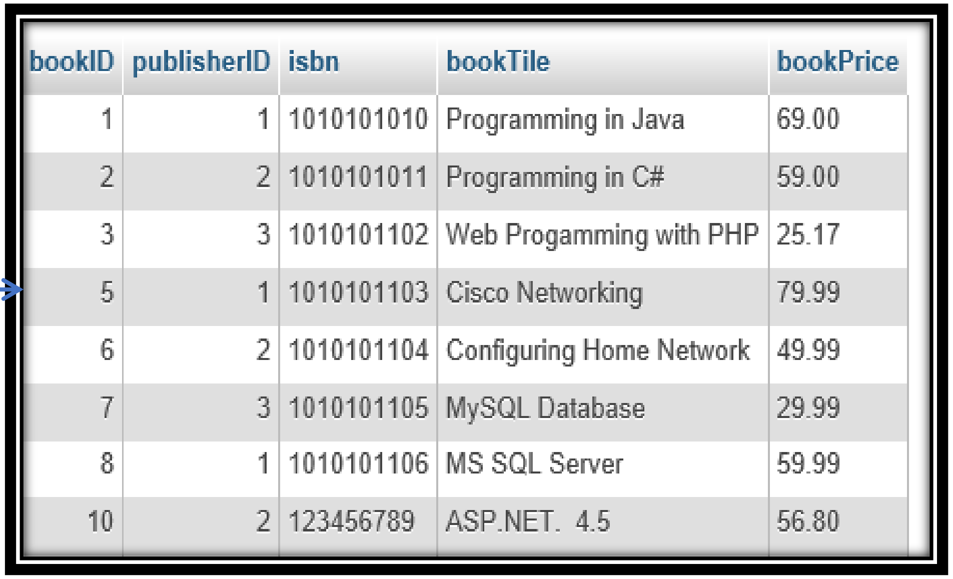
ISBN	Book Title	Book Price
1010101010	Programming in Java	69.00
1010101103	Cisco Networking	79.99
1010101106	MS SQL Server	59.99

Two Database Tables



publisherID	publisherName
1	Apress
2	Microsoft Press
3	Sams Publishing

Publishers



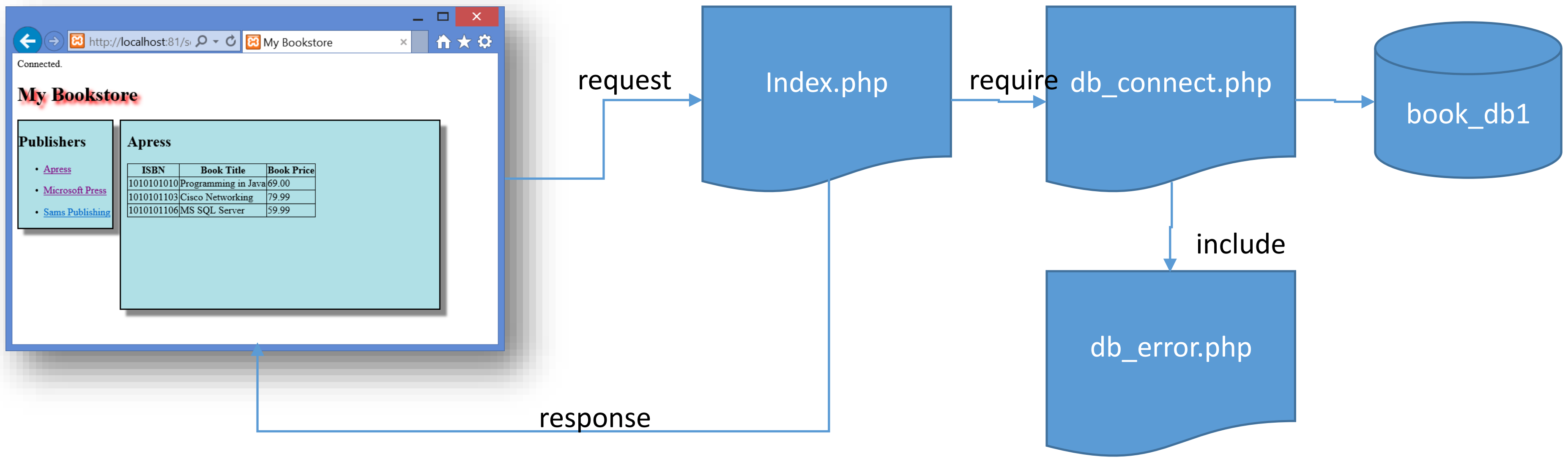
bookID	publisherID	isbn	bookTile	bookPrice
1	1	1010101010	Programming in Java	69.00
2	2	1010101011	Programming in C#	59.00
3	3	1010101102	Web Progamming with PHP	25.17
5	1	1010101103	Cisco Networking	79.99
6	2	1010101104	Configuring Home Network	49.99
7	3	1010101105	MySQL Database	29.99
8	1	1010101106	MS SQL Server	59.99
10	2	123456789	ASP.NET. 4.5	56.80

Books

Create Table Script

```
CREATE DATABASE book_db1;

USE book_db1;
CREATE TABLE publishers (
  publisherID      INT(11)       NOT NULL   AUTO_INCREMENT,
  publisherName     VARCHAR(255)   NOT NULL,
  PRIMARY KEY (publisherID)
);



CREATE TABLE books (
  bookID          INT(11)       NOT NULL   AUTO_INCREMENT,
  publisherID      INT(11)       NOT NULL,
  isbn            VARCHAR(20)    NOT NULL   UNIQUE,
  bookTile         VARCHAR(255)   NOT NULL,
  bookPrice        DECIMAL(10,2)  NOT NULL,
  PRIMARY KEY (bookID)
);
```

The Structure of Search-Book Application



Connected.

My Bookstore

Publishers

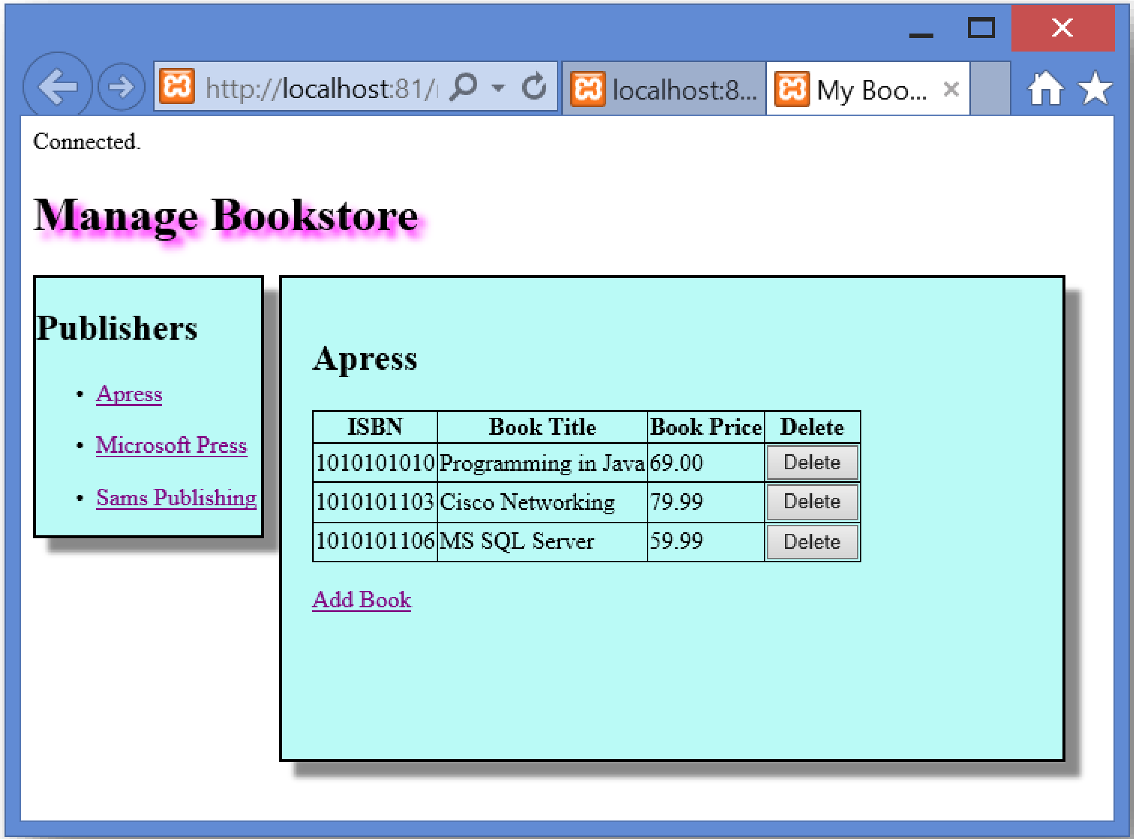
- Apress
- Microsoft Press
- Sams Publishing

Apress

ISBN	Book Title	Book Price
1010101010	Programming in Java	69.00
1010101103	Cisco Networking	79.99
1010101106	MS SQL Server	59.99

The Manage-Book Application

- Lets the administrator to add and delete books



Connected.

Manage Bookstore

Publishers

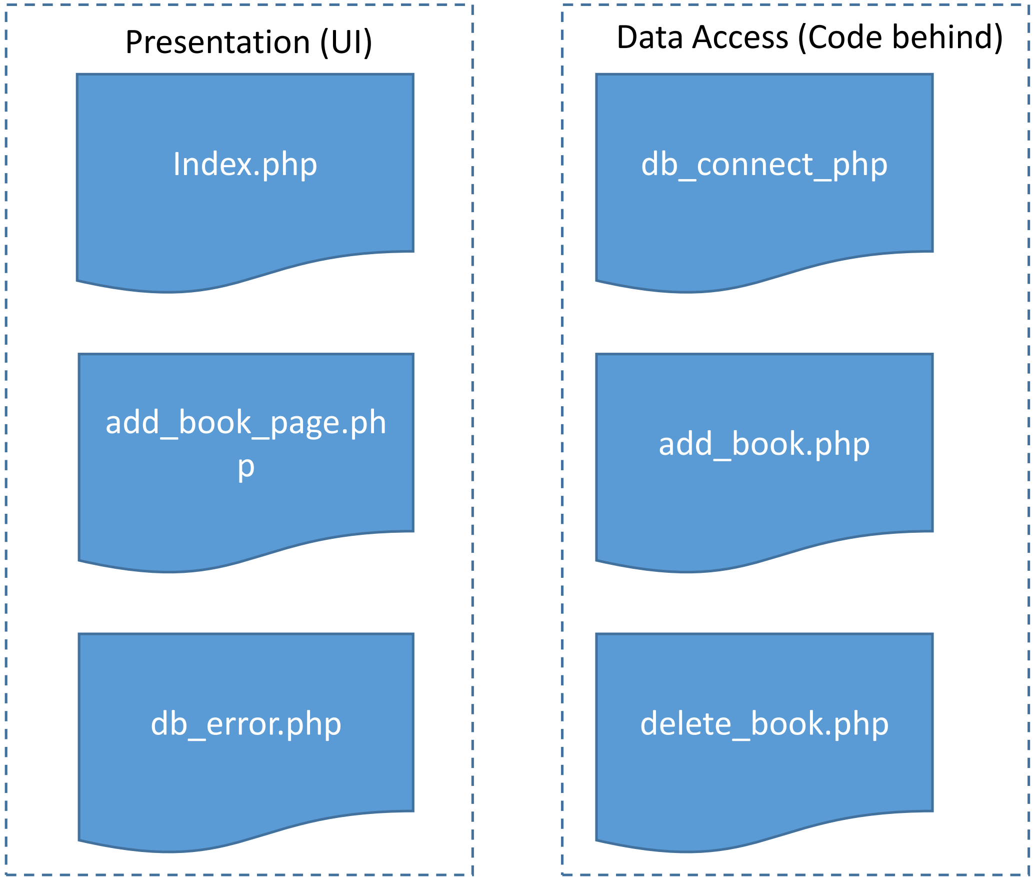
- Apress
- Microsoft Press
- Sams Publishing

Apress

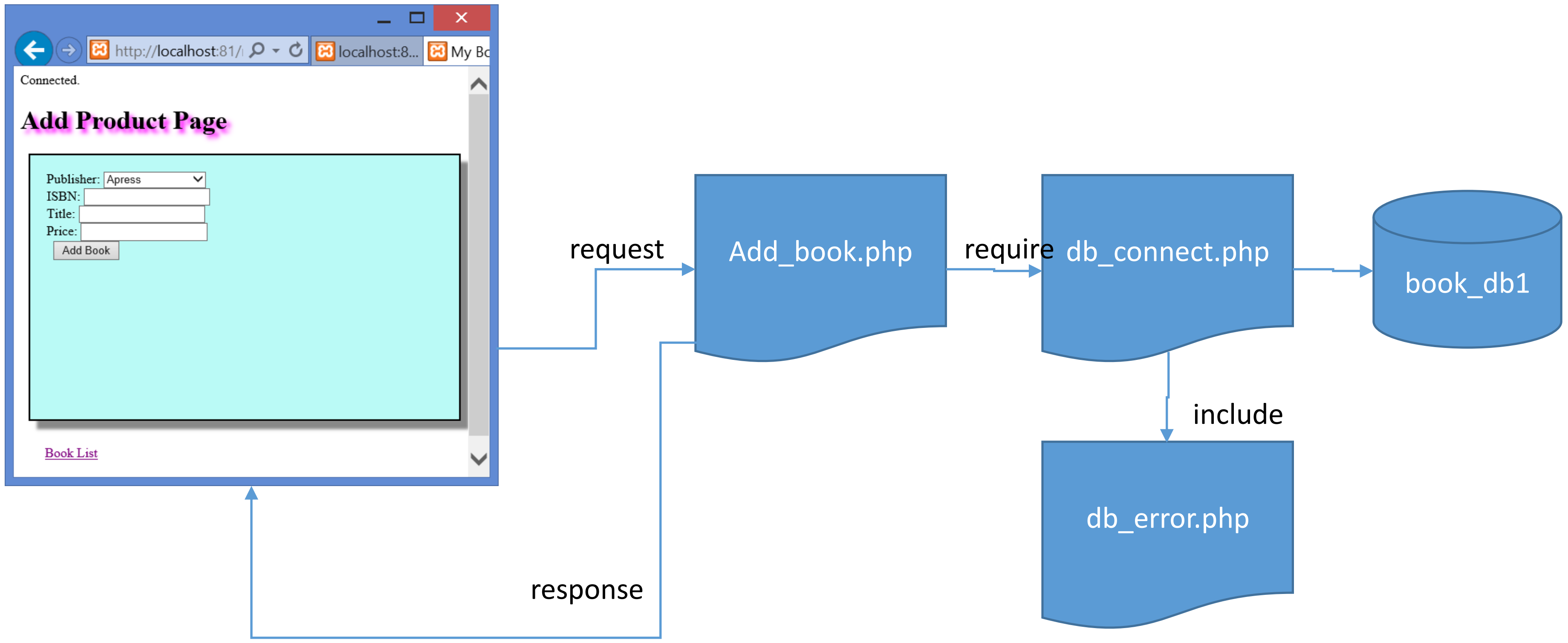
ISBN	Book Title	Book Price	Delete
1010101010	Programming in Java	69.00	Delete
1010101103	Cisco Networking	79.99	Delete
1010101106	MS SQL Server	59.99	Delete

Add Book

The Structure of Manage-Book Application



Adding a new book


http://localhost:81/
localhost:8...
My Bc
Connected.
Add Product Page
Publisher: Apress
ISBN:
Title:
Price:
Add Book
Book List
request
Add_book.php
require
db_connect.php
book_db1
include
db_error.php
response