CRUD Operation using PHP & Mongodb

In this section, we are going to perform view, insert, delete and update operations. We will use PHP and Mongodb to do this. The database which we used in our application is MongoDB. The step-by-step process to create, update, delete modules is described as follows. In PHP, the various operation of MongoDB can be easily used by the given example, like update, add, find, search, delete, select, etc. The connection between MongoDB and PHP will also be learned in this example.

The open-source and very famous database is MongoDB. It is a NoSQL database that is based on the document. We will use the MongoDB database in our application if the database has more data or it contains a large number of data. MongoDB database stores data by using less memory, and it can also be used to fetch quick records. The steps to do this are described as follows:

Step 1

In this step, we are going to Create MongoDB Database. In our example, we need to create our book collection and MongoDB database. When we successfully install the MongoDB database, we will use our command prompt, connect to MongoDB. For connection, we will create a database and then a collection. After that, we will insert the book by using the following command like this:

```
mongo
> use hddatabase
> db.books.insert( { "name": "laravel", "detail": "test" } )
```

Step 2:

In this step, we are going to Install MongoDB and MongoDB libraries. In our application, we will use composer package manager so that we can install it. For this, we will create a folder in our root directory. After that, we will open our command prompt and run the following command:

```
composer require mongodb/mongodb
```

Step 3:

In this step, we are going to Create Config File for CRUD App. After creating the config file, we will use it to provide a connection with MongoDB. The code to create the file and make a connection is described as follows. For this, we also want to set username, port, password, and URL. We will set the collection name and database also. We will describe "books" as the collection name and "hddatabase" as the database.

config.php

```
<?php

require_once __DIR__ . "/vendor/autoload.php";

$collection = (new MongoDB\Client)->hddatabase->books;

?>
```

Step 4:

In this step, we are going to Create an Index, Create, Edit, and Delete files. For this, we will create many files, such as edit.php, index.php, create.php, and delete.php. The command to create these files is described as follows:

index.php

```
<?php
   session_start();
?>
<!DOCTYPE html>
<html>
<head>
   <title> CRUD Operation using MongoDB and PHP </title>
   <link href="https://stackpath.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
rel="stylesheet" crossorigin="anonymous">
</head>
<body>

<div class="container">
<h1> CRUD Operation using MongoDB and PHP </h1>

<a href="create.php" class="btn btn-success">Add Book</a>

<?php

   if(isset($_SESSION['success'])){
      echo "<div class='alert alert-success'>".$_SESSION['success']."</div>";
   }

?>

<table class="table table-borderd">
   <tr>
      <th>Name</th>
      <th>Details</th>
      <th>Action</th>
   </tr>
   <?php

      require 'config.php';

      $books = $collection->find([]);

      foreach($books as $book) {
         echo "<tr>";
         echo "<td>".$book->name."</td>";
         echo "<td>".$book->detail."</td>";
         echo "<td>";
         echo "<a href='edit.php?id=".$book->_id."' class='btn btn-primary'>Edit</a>";
         echo "<a href='delete.php?id=".$book->_id."' class='btn btn-danger'>Delete</a>";
         echo "</td>";
         echo "</tr>";
      };

   ?>
</table>
</div>

</body>
</html>
```

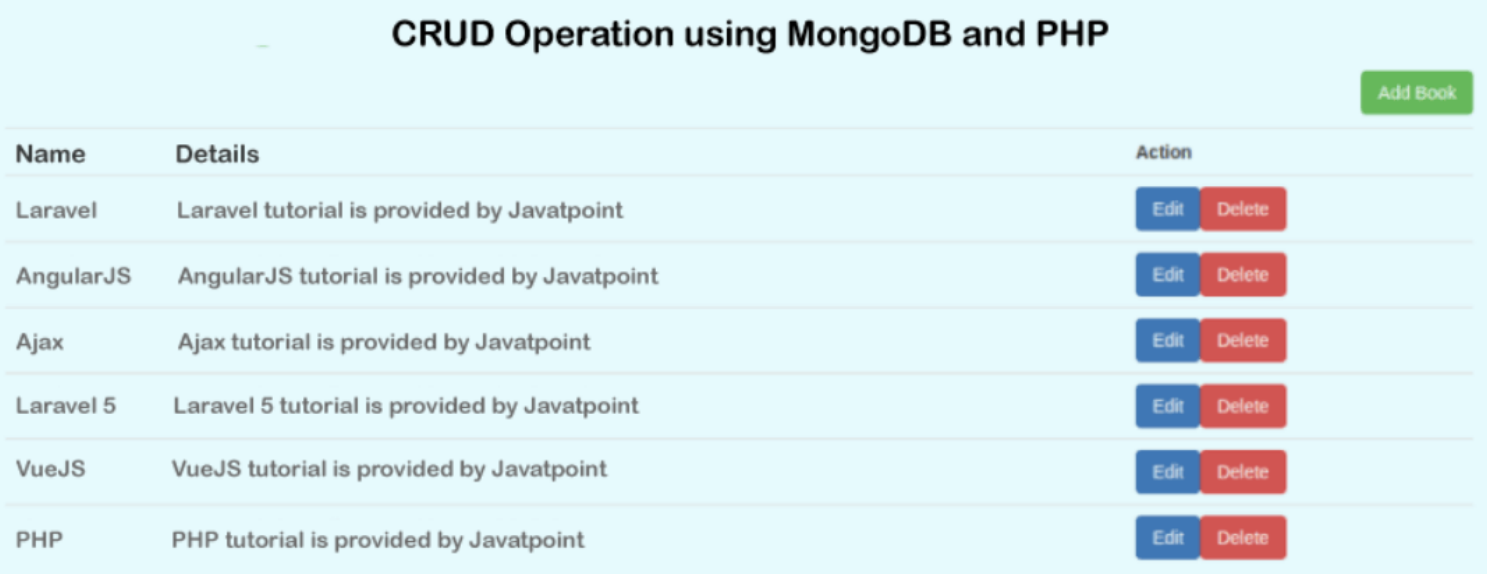
create.php

```
<?php

session_start();

if(isset($_POST['submit'])){

   require 'config.php';

   $insertOneResult = $collection->insertOne([
       'name' => $_POST['name'],
       'detail' => $_POST['detail'],
   ]);

   $_SESSION['success'] = "Creation of Book is successful";
   header("Location: index.php");
}

?>

<!DOCTYPE html>
<html>
<head>
   <title> CRUD Operation using MongoDB and PHP </title>
   <link href="https://stackpath.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
rel="stylesheet" crossorigin="anonymous">
</head>
<body>

<div class="container">
   <h1>Create Book</h1>
   <a href="index.php" class="btn btn-primary">Back</a>

   <form method="POST">
      <div class="form-group">
         <strong>Name:</strong>
         <input type="text" name="name" required="" class="form-control" placeholder="Name">
      </div>
      <div class="form-group">
         <strong>Detail:</strong>
         <textarea class="form-control" name="detail" placeholder="Detail"
placeholder="Detail"></textarea>
      </div>
      <div class="form-group">
         <button type="submit" name="submit" class="btn btn-success">Submit</button>
      </div>
   </form>
</div>

</body>
</html>
```

edit.php

```
<?php

session_start();

require 'config.php';

if (isset($_GET['id'])) {
   $book = $collection->findOne(['_id' => new MongoDB\BSON\ObjectID($_GET['id'])]);
}

if(isset($_POST['submit'])){

   $collection->updateOne(
       ['_id' => new MongoDB\BSON\ObjectID($_GET['id'])],
       ['$set' => ['name' => $_POST['name'], 'detail' => $_POST['detail'],]]
   );

   $_SESSION['success'] = "Updating of Book is successful";
   header("Location: index.php");
}

?>

<!DOCTYPE html>
<html>
<head>
   <title> CRUD Operation using MongoDB and PHP </title>
   <link href="https://stackpath.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
rel="stylesheet" crossorigin="anonymous">
</head>
<body>

<div class="container">
   <h1>Create Book</h1>
   <a href="index.php" class="btn btn-primary">Back</a>

   <form method="POST">
      <div class="form-group">
         <strong>Name:</strong>
         <input type="text" name="name" value="<?php echo $book->name; ?>" required=""
class="form-control" placeholder="Name">
      </div>
      <div class="form-group">
         <strong>Detail:</strong>
         <textarea class="form-control" name="detail" placeholder="Detail" placeholder="Detail"><?
php echo $book->detail; ?></textarea>
      </div>
      <div class="form-group">
         <button type="submit" name="submit" class="btn btn-success">Submit</button>
      </div>
   </form>
</div>

</body>
</html>
```

Now our above code is ready, and we can run it on our local. When we run this, the following output will be generated:



CRUD Operation using MongoDB and PHP

Add Book

Name	Details	Action
Laravel	Laravel tutorial is provided by Javatpoint	Edit Delete
AngularJS	AngularJS tutorial is provided by Javatpoint	Edit Delete
Ajax	Ajax tutorial is provided by Javatpoint	Edit Delete
Laravel 5	Laravel 5 tutorial is provided by Javatpoint	Edit Delete
VueJS	VueJS tutorial is provided by Javatpoint	Edit Delete
PHP	PHP tutorial is provided by Javatpoint	Edit Delete

← Prev

Next →

Feedback

- Send your Feedback to [email protected]

Help Others, Please Share

R Programming tutorial

R Programming

RxJS tutorial

RxJS

React Native tutorial

React Native

Python Design Patterns

Python Design Patterns

Python Pillow tutorial

Python Pillow

Python Turtle tutorial

Python Turtle

Keras tutorial

Keras

Preparation

Aptitude

Aptitude

Logical Reasoning

Reasoning

Verbal Ability

Verbal Ability

Interview Questions

Interview Questions

Company Interview Questions

Company Questions

Trending Technologies

Artificial Intelligence Tutorial

Artificial Intelligence

AWS Tutorial

AWS

Selenium tutorial

Selenium

Cloud Computing tutorial

Cloud Computing

Hadoop tutorial

Hadoop

ReactJS Tutorial

ReactJS

Data Science Tutorial

Data Science

Angular 7 Tutorial

Angular 7

Blockchain Tutorial

Blockchain

Git Tutorial

Git

Machine Learning Tutorial

Machine Learning

DevOps Tutorial

DevOps

B.Tech / MCA

DBMS tutorial

DBMS

Data Structures tutorial

Data Structures

DAA tutorial

DAA

Operating System tutorial

Operating System

Computer Network tutorial

Computer Network

Compiler Design tutorial

Compiler Design

Computer Organization and Architecture

Computer Organization

Discrete Mathematics Tutorial

Discrete Mathematics

Ethical Hacking Tutorial

Ethical Hacking

Computer Graphics Tutorial

Computer Graphics

Software Engineering Tutorial

Software Engineering

html tutorial

Web Technology

Cyber Security tutorial

Cyber Security

Automata Tutorial

Automata

C Language tutorial

C Programming

C++ tutorial

C++

Java tutorial

Java

.Net Framework tutorial

.Net

Python tutorial

Python

List of Programs

Programs

Control Systems tutorial

Control System

Data Mining Tutorial

Data Mining

Data Warehouse Tutorial

Data Warehouse