Master in PHP with Laravel

About
DevOpsSchool

DevOpsSchool is a unit of "Cotocus PVT ltd" and a leading platform which helps IT organizations and professionals to learn all the emerging technologies and trend which helps them to learn and embrace all the skills, intelligence, innovation and transformation which requires to achieve the end result, quickly and efficiently. We provide over 40 specialized programs on DevOps, Cloud, Containers, Security, AI, ML and on Big data that are focused on industry requirement and each curriculum is developed and delivered by leading experts in each domain and aligned with the industry standards.

About Course

PHP is a great beginner programming language for anyone who is wanting to cross over into the coding world. Though easy to learn, it is an extremely robust language and can be used to create large scale websites. You might not be aware that some of the most popular websites are built with PHP. These include Yahoo, Tumblr, Wikipedia, and Facebook.

In this course, we are going to start with the basics and work our way to creating a fully interactive website utilizing the PHP language. Some of the major topics that we will cover include PHP syntax, built-in functions, persisting data across web pages and connecting to a database. I hope you'll on this journey to learn PHP with the PHP Fundamentals course at DevOpsSchool.com. In addition you can also learn Laravel course.

Co-coordinator – Akanksha Kumari

Call/WhatsApp: - +91 1800 889 7977

Mail Address: -

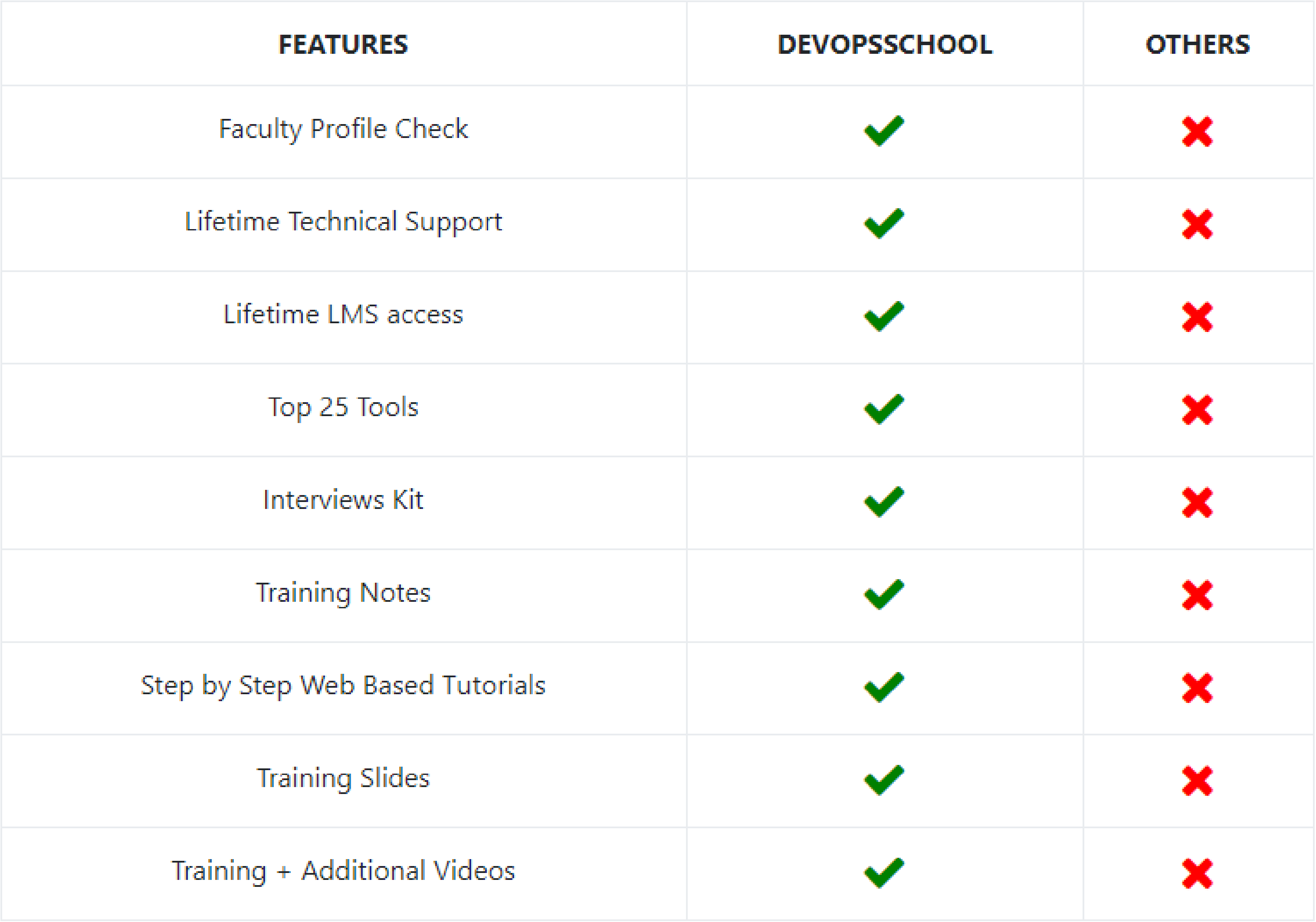
contact@DevOpsSchool.com

Secondary contact – Patrick

Call/WhatsApp: - +91 7004 215 841

Mail Address: -contact@DevOpsSchool.com

Duration	80 Hours
Mode	Online (Instructor-led, live & Interactive)
Projects (Real time scenario based)	1



FEATURES	DEVOPSSCHOOL	OTHERS
Faculty Profile Check	✓	✗
Lifetime Technical Support	✓	✗
Lifetime LMS access	✓	✗
Top 25 Tools	✓	✗
Interviews Kit	✓	✗
Training Notes	✓	✗
Step by Step Web Based Tutorials	✓	✗
Training Slides	✓	✗
Training + Additional Videos	✓	✗

AGENDA OF THE MASTER IN PHP WITH LARAVEL

Introduction - PHP

- Introduction
- Intended Audience
- History of PHP
- What's Possible?
- Sample Programs
- Reserved Keywords

General/Data Type

- What are the basic commands in Selenium?
- Intended Audience
- History of PHP
- What's Possible?
- Sample Programs
- Reserved Keywords

Functions

- What Is a Function?
- Creating a Function
- Calling a Function
- Passing Parameters
- Default Parameters
- Returning Values
- Variable Functions
- Variable Scope
- Global Variables

String

- Single Quoted Strings
- Double Quoted Strings
- Here Document
- print()
- Changing Case
- strlen()
- strpos()
- str_replace()
- substr()
- str_split()

Arrays

- overview
- About Arrays
- Indexed Arrays
- Associative Arrays
- array_key_exists()
- in_array()
- array_push()
- array_pop()
- unset()
- Sorting Arrays
- count()
- Foreach Loop
- Multi-dimensional Array

Classes & Object

- Creating a Simple Class
- Creating an Object
- Creating Properties
- Accessing Properties
- Creating Constants
- Accessing Constants
- Creating Methods
- $this
- Accessing Methods
- Constructors
- Initial Parameter Passing
- Inheritance
- Protected Properties
- Private Properties
- Static Properties & Methods
- Scope Resolution Operator ::
- Include File
- Require File

Operators and Control Structures

- Arithmetic Operators
- Incrementing/Decrementing Operators
- Assignment Operators
- String Operator
- Comparison Operators
- Spaceship Operator
- Logical Operators
- If Statement
- Else If
- Switch Statement
- Ternary Operator
- Null Coalesce Operator
- While Loop
- For Loop

Database

- Possible Databases
- PDO vs. MySQLi
- Setting up MySQL on Windows
- Creating a Database and Table with phpMyAdmin
- Connecting to Database
- Close Database Connection
- Executing a Query - Delete, Update, Insert
- Grabbing Inserted Id
- Select Data From Database
- Prepared Statement Example
- PDO Example

Web Programming

- Setup Web Site Location - Linux
- Setup Web Site Location - Windows
- Web Site Structure and Layout
- Shorten Syntax with HTML
- $_GET
- $_POST
- Form Validation
- session_start()
- $_SESSION
- session_destroy()

PHP - TOAST

- Setup Web Site Location - Linux
- Setup Web Site Location - Windows
- Web Site Structure and Layout
- Shorten Syntax with HTML
- $_GET
- $_POST
- Form Validation
- session_start()
- $_SESSION
- session_destroy()

Building the LAMP Stack

- Introduction
- Installation
- Alternatives
- Accessing Form Data with PHP
- HTML forms
- Demo: Creating a Form
- Accessing Form Data with PHP
- Validating User Input
- Demonstration -- Input Validation

The Lending Library's Database

- Designing the Database
- Creating and Populating the Database
- Querying and Updating the Database
- Accessing a Database from PHP

Introducing PDO

- Connecting to the Database
- Book Search Demonstration
- Quoting and Input Validation
- Mysqli

Doing More with the Database

- Using Prepared Statements
- Non-queries and Summary Functions
- Creating and Using Stored Procedures and Functions

Maintaining State in Web Applications

- The Problem of Maintaining State
- PHP Sessions
- Using Hidden Fields
- Using Cookies
- The Library: User Preferences and Shopping Carts

RESTfull Web Services With PHP and Laravel

- Introduction
- Course Structure
- What are RESTful Services?
- Demo Overview
- Postman Setup & Demo
- Demo: Course Project
- Refresher on RESTful Services
- Refresher on Laravel & Setting Laravel Up
- Demo: Course Project Setup
- Cross-origin Resource Sharing (CORS)
- Wrap Up

Introduction - Laravel

- What is Laravel?
- MVC vs Non-MVC
- MVC and the Structure of This Course
- Setting up a Laravel Project
- Understanding the Folder Structure
- A Quick Intro to the Artisan Command Line Interface

Views & Blade Templating Engine

- Introduction
- Views and Templating Engines
- Why Use Templating Engines?
- Blade - Displaying Data and Using Layouts
- Demo:Using Layouts
- Demo:Displaying Data
- Blade - Partials
- Demo: Partials
- Blade - Control Stuctures
- Demo: Control Stuctures
- Blade - XSS Protection
- Demo: XSS Protection
- Facades

Views & Blade Templating Engine

- Introduction
- Views and Templating Engines
- Why Use Templating Engines?
- Blade - Displaying Data and Using Layouts
- Demo:Using Layouts
- Demo:Displaying Data
- Blade - Partials
- Demo: Partials
- Blade - Control Stuctures
- Demo: Control Stuctures
- Blade - XSS Protection
- Demo: XSS Protection
- Facades

Working with Routes

- Introduction
- Views and Templating Engines
- Why Use Templating Engines?
- Blade - Displaying Data and Using Layouts
- Demo:Using Layouts
- Demo:Displaying Data
- Blade - Partials
- Demo: Partials
- Blade - Control Stuctures
- Demo: Control Stuctures
- Blade - XSS Protection
- Demo: XSS Protection
- Facades

Handaling Requests & Responses

- Introduction
- Handling Requests and Sending Responses
- Demo: Extracting Data from a GET Request
- Demo: Sending Data to a View
- Handling POST Requests with Dependency Injection
- Dependency Injection vs Facades
- Demo:Using Dependency Injection
- Protecting Against CSRF Attacks
- Demo:CSRF Protection
- Using Sessions to Show Temporary Data
- A Case for User Input Validation
- Demo:Implementing Input Validation
- Demo:Showing Validation Errors
- Demo:Outsourcing Error Messages into Partials

Configuration & Environments

- Configuration and Environments Overview
- Configuring the Database Connection

Using Controllers and Models

- Introduction
- Controllers and Models - Overview
- Demo:Creating the Post Model
- Demo:Adding Methods to the Model
- Demo: Creating the Post Controller
- Connecting Controller Actions and Routes
- How laravel Finds the Controller
- Alternative Route Syntax
- Demo:Finishing the Post Model
- Demo:Finishing the Controller
- using Validation in Controller
- Finishing the Course Project

Databases - Introduction & Migration

- Databases Bird's Eye Overview
- Migrations Overview
- Creating and Running Migrations

Databeses - Eloquent ORM

- Understanding Eloquent ORM Models
- Models- Creating and Updating,Introducing Tinker
- Models - Retrieving Single Model
- Models- Retrieving Multiple Models and Collections Overview
- Models- Using the Query Builder
- Practical - List of Blog Posts and Single Blog Post

Forms(Markup,CSRF,Validation,Errors,Flash Messages,Mass Assignment)

- Forms Markup
- Cross Site Request Forgery Explained
- Forms - Storing Submitted Data
- Forms - Input Validation
- Forms - Displaying Validation Errors
- Forms - Form Request Classes
- Session Flash Messages
- Forms - Old Input Helper
- Forms/Models - Model Mass Assignment

- CRUD - Edit Form
- CRUD - Update Action
- CRUD - Deleting Using Forms
- CRUD - Deleting Models

Assets & Styling(Javascript,CSS, Laravel Mix, Bootstrap)

- Introduction to Laravel Mix
- Installing Bootstrap
- Using NPM and Compiling Assets With Mix/Webpack
- Including Assets in Views
- Versioned Assets(Cache Improvements)
- Introduction to Bootstrap CSS
- Understanding CSS Flexbox in General and in Bootstrap
- Layout Grid and Styling Header Bar
- Styling Forms
- Styling Post List Page
- Styling Single Post Page
- Styling Flash Messages and Error Messages

Testing Basics

- Testing
- Testing configuration and environment
- Writing first function test
- Testing database interactions
- Testing store() action (model creation)
- Testing for failure
- Testing update() action (model updates)
- Testing delete() action (model deletion)

One to One Eloquent Relations

- One to One relation with migration
- One to One assigning relationship
- One to One querying relationship

One to Many Eloquent Relations

- One to Many relation with migration
- One to Many assigning relationship
- One to Many querying relationship

Querying Basics

- Lazy Loading vs Eager Loading
- Querying Relationship existence
- Querying relationship absense
- Counting related models
- Using withCount() in practice (fetching count of comments) with test

Model Factories

- Lazy Loading vs Eager Loading
- Querying Relationship existence
- Querying relationship absense
- Counting related models
- Using withCount() in practice (fetching count of comments) with test

Querying Basics

- Lazy Loading vs Eager Loading
- Querying Relationship existence
- Querying relationship absense
- Counting related models
- Using withCount() in practice (fetching count of comments) with test

Model Factories

- Model Factory Introduction
- Model Factory States
- Model Factory callbacks(afterCreating, afterMaking)
- Application:Implementing comment list

Authentication

- Authentication Overview
- How user registration works in laravel
- Guard component and how logging users in works
- Custom registration from and Auth routes
- Formatting validation errors
- RedirectIfAuthenticated middleware
- Log-in form with "Remember Me" feature
- Logging out.@guest directive,debugging CSRF token errors
- Retrieving the currently authenticated user
- Protecting routes(requiring authentication)
- Testing routes that requires

Database Seeding

- Refreshing database, database foreign keys and existing data
- Problem: SQLite test database NOT NULL problem
- Database seeding basics
- Using Model Factory inside Seeder
- Model relations inside seeder
- Individual seeder classes
- Making seeder interactive

Deleting Models and Soft Deletes

- Deleting related model using model events
- Deleting related models using cascading
- Soft deletes
- Soft deletes querying
- Restoring soft deleted model
- Testing soft deleted models

Authorization, Policies, Gates

- Authorization introduction
- Introduction to Gates
- Using authorize() helper
- Verifying permissions of the user
- Admin users and overriding permissions
- Policies introduction
- Policy or Gate?
- Verifying permissions in Blade templates
- Using middleware to authorize routes
- Application: updating tests

Query Scopes - Local & Global

- Application: setting user_id for the new BlogPost
- Global Query Scopes introduction
- Global Query Scopes and potential issues
- Local Query Scopes introduction
- Practical: Local Query Scope - most commented posts
- Practical: Local Query Scope - most active users
- Practical: Local Query Scope - most active users last month
- Practical: Global Query Scope - admin can see deleted posts

Laravel Blade Components

- Blade Components introduction
- Component aliases
- Conditional rendering in Component
- Practical: creating reusable component for dates
- Complicated example of conditional rendering
- Application: Fixing an issue with HAVING clause

Caching

- Caching introduction
- Laravel Debugbar
- Storing data in cache
- Removing from cache
- Cache facade
- Practical: using cache as storage
- Practical: using cache for storage implementation
- Using and setting up Redis as cache storage
- Cache tags introduction
- Practical: using cache tags

Many to Many Eloquent Relations

- Caching introduction
- Laravel Debugbar
- Storing data in cache
- Removing from cache
- Cache facade
- Practical: using cache as storage
- Practical: using cache for storage implementation
- Using and setting up Redis as cache storage
- Cache tags introduction
- Practical: using cache tags

Reusable Components, query Scopes, Route Model Binding)

- Practical: User to Comment OneToMany relation and migration
- Practical: comments form and reusable errors component
- Route Model Binding
- Eager loading nested relationships
- Converting repeating queries to query scopes

File Storage and Uploading

- File Storage introduction
- File upload form
- Handling file uploads
- Using Storage facade to store files
- Getting the URL of stored file
- Practical: Image model, OneToOne relation and migrations
- Uploaded image URL
- Practical: Displaying uploaded image and styling
- Deleting files
- Validating uploaded files (size, type, dimensions)

One to One Polymorphic Eloquent Relation

- Section intrduction
- Practical: Scaffolding UserController and UserPolicy, using authorizeResource
- Practical: Views for showing/editing user profile
- OneToOne Polymorphic explained
- OneToOne Polymorphic migration
- OneToOne Polymorphic defining relation
- OneToOne Polymorphic associating
- Practical: OneToOne Polymorphic with BlogPost and Image

One to Many Polymorphic Eloquent Relation

- OneToMany Polymorphic explained
- OneToMany Polymorphic migration & relation
- OneToMany Polymorphic associating
- Practical: OneToMany Polymorphic views
- Practical: Running tests on MySQL database
- OneToMany Polymorphic seeder

Thank you!

Connect with us for more info

Call/WhatsApp: - +91 968 682 9970

Mail: - contact@DevOpsSchool.com

www.DevOpsSchool.com