



Laravel

Web Application
Development with PHP &
Laravel [80 Hrs]

Instructor: Pondit Team

E-Mail:

Phone:

Overview

আইটি ইন্ডাস্ট্রির জন্য দক্ষ জনশক্তি তৈরিতে পন্ডিট স্পন্সরশীপ প্রদান করছে। যার পরিপ্রেক্ষিতে ইউনিভার্সিটির ছাত্রছাত্রীরা একটি পরীক্ষায় ভালো রেজাল্টের মাধ্যমে ১০০% পর্যন্ত ফ্রি কোর্স করার সুযোগ পাচ্ছেন।

পরীক্ষার মাধ্যমে আপনি ১০০% পর্যন্ত স্পন্সরশীপ পাবেন তবে আপনাকে অবশ্যই পরীক্ষায় ৮০% মার্ক্স পেতে হবে। ১০০% স্পন্সরশীপ না পেলে পরীক্ষার রেজাল্টের উপর ভিত্তি করে ৭৫% অথবা ৫০% ডিসকাউন্ট পাবেন।

পিএইচপি এবং লারাভেল কেন শিখবেন?

- ১। পিএইচপি এবং লারাভেল এর জব মার্কেট অনেক বড়।
- ২। শেখার ইচ্ছা থাকলে এবং পরিশ্রম করলে খুব সহজেই শেখা যায়।
- ৩। এছাড়া PHP তে প্রায় হাজারের বেশী বিল্ট-ইন ফাংশন রয়েছে, যাতে করে অনেক জটিল কাজ ডেভেলপার একটি ফাংশন ব্যবহার করেই সমাধান করতে পারে। এ সহজতার কারণে PHP খুব তাড়াতাড়ি শেখা যায়।
- ৪। Back End বা (সার্ভার সাইড) ওয়েব ডেভেলপমেন্টে এর জন্য পিএইচপি অনেক আগে থেকেই বেশ জনপ্রিয়। Facebook, Yahoo, Wikipedia, Wordpress এর মতো বড় বড় প্রতিষ্ঠান তাদের মূল ওয়েবসাইট সহ বিভিন্ন কাজে PHP ব্যবহার করে আসছে। এছাড়া বর্তমানে ওয়েব টেকনোলজির ৭৪% জায়গা পিএইচপির দখলে রয়েছে।
- ৫। ক্যারিয়ার হিসেবে PHP'র এখন বেশ চাহিদা রয়েছে। যারা নিজেকে ওয়েব ডেভেলপার হিসেবে দেখতে চান, তারা খুব সহজে অল্প সময়ে একজন PHP ডেভেলপার হতে পারেন। বর্তমানে PHP'র ফ্রেমওয়ার্কগুলোর মধ্যে অন্যতম লারাভেল(Laravel) এর কথা না বললেই নয় এর ব্যবহার ও চাহিদার ব্যাপকতা; ক্যারিয়ারে পিএইচপির সুনিশ্চিত ভবিষ্যতের কথা মনে করিয়ে দেয়।

পন্ডিট - এ কেন ড্রেইনিং করবেন?

- ১। দক্ষ ড্রেইনারে মাধ্যমে শিখতে পারবেন।
- ২। কোর্স সম্পূর্ণ করার পরে দক্ষতার ভিত্তিতে জব এর ব্যবস্থা করবে পন্ডিট।
- ৩। আমাদের নিজস্ব ভার্সুয়াল প্ল্যাটফর্মের মাধ্যমে ক্লাস নেয়া হয়।
- ৪। যা শিখানো হবে তার উপরে নিয়মিত পরীক্ষা নেয়া হবে।

Audience

- শিক্ষার্থী (যারা পড়াশুনা শেষ করেই জব করতে চান অথবা যারা পড়াশুনা করা অবস্থাতেই জব করতে চান)

Prerequisites

পরিশ্রমী হতে হবে, প্রোগ্রামিং শেখার আগ্রহ থাকতে হবে।

Course Information

- Duration : 80 Hours
- Day : শনিবার এবং বুধবার
- Time : রাত ৯.০০ টা থেকে ১১.০০ টা পর্যন্ত
- Reg. Start : Mar 1, 2022
- Reg. End : May 27, 2022
- Class Start : Jun 4, 2022

Evaluation

Curriculum


1. PHP

- 1.1. Course Overview
- 1.2. Local Development Environment Tools
- 1.3. Variables in PHP
- 1.4. Primitive Data Types
- 1.5. Defining Constants and Variables
- 1.6. Logical Operators
- 1.7. Relational Operators
- 1.8. Bitwise Operators
- 1.9. Other Operators
- 1.10. PHP7 Null Coalescing operator
- 1.11. PHP7 Spaceship Operator
- 1.12. What are Arrays?
- 1.13. Usage of Arrays in PHP
- 1.14. Array Indexing
- 1.15. Initializing Arrays
- 1.16. Operating on Arrays
- 1.17. Sorting Arrays
- 1.18. One-Dimensional Arrays
- 1.19. Multi-Dimensional Arrays
- 1.20. Associative Arrays
- 1.21. Array Functions
- 1.22. True and False Expressions
- 1.23. if, else and elseif
- 1.24. switch/case Statement
- 1.25. The ? (Ternary) Operator
- 1.26. While loop
- 1.27. Do... while loop
- 1.28. For loop
- 1.29. Exit and break
- 1.30. Foreach loop
- 1.31. Introduction to Functions
- 1.32. Declaring Functions
- 1.33. Scope
- 1.34. Passing Arguments to Functions
- 1.35. Local and Global Scope
- 1.36. Passing Arguments to Functions
- 1.37. Variable Scoping and Return Values
- 1.38. Recursion
- 1.39. Optional and REST Parameters
- 1.40. Argument Unpacking
- 1.41. Generator Return Expressions
- 1.42. Dynamic Function Calls
- 1.43. Predefined PHP Functions
- 1.44. Closure
- 1.45. Checking for Form Submission
- 1.46. Extracting Information from form

- 1.47. Validating The Form Values
- 1.48. Make a simple GPA calculator
- 1.49. Defining a class
- 1.50. Defining methods
- 1.51. Instantiating a Class
- 1.52. Defining properties
- 1.53. Access Control Modifiers
(public/private/protected)
- 1.54. Static Modifier
- 1.55. Getters and Setters
- 1.56. Referencing Parent Class with Static
- 1.57. Constructors and Destructors
- 1.58. Convert GPA calculator to OOP.
- 1.59. PHP7-Anonymous class
- 1.60. Abstract class
- 1.61. Interface
- 1.62. Final class/method
- 1.63. Polymorphism
- 1.64. Trait
- 1.65. Namespace
- 1.66. PSR-1 (Basic Coding Standard)
- 1.67. PSR-4 (Autoload Standard)
- 1.68. Gitlab
- 1.69. Github
- 1.70. MySQL as a Client/Server Solution
- 1.71. Introduction to MySQL Capabilities as a
Powerful RDBMS
- 1.72. Installing and Configuring MySQL
- 1.73. Connecting to MySQL
- 1.74. PHP Functions Specific to MySQL
- 1.75. Executing SQL Calls
- 1.76. SELECT
- 1.77. INSERT
- 1.78. FETCH
- 1.79. UPDATE
- 1.80. DELETE
- 1.81. Using PHP MyAdmin to Configure MySQL
- 1.82. Create
- 1.83. Read
- 1.84. Update
- 1.85. Delete
- 1.86. Validation
- 1.87. Error handling
- 1.88. Flush message (session)

2. Laravel

- 2.1. Why Framework?
- 2.2. Brief concept on Laravel
- 2.3. About MVC Design Pattern.
- 2.4. Installation
- 2.5. Configuration



- 2.6. Directory Structure
- 2.7. Q & A – Open discussion on outline
- 2.8. Basic Routing
- 2.9. Route Parameters
- 2.10. Named Routes
- 2.11. Route Groups
- 2.12. Creating Controllers
- 2.13. Routing Controllers
- 2.14. Passing data
- 2.15. Views & Blade Templating
- 2.16. Template Inheritance
- 2.17. Extending A Layout
- 2.18. Displaying Data
- 2.19. Environment configurations.
- 2.20. Generating Migrations.
- 2.21. Migration Structure.
- 2.22. Creating migration and dropping them
- 2.23. Adding columns to existing tables using migrations
- 2.24. Rollback, refresh, reset
- 2.25. Modifying Columns
- 2.26. Foreign Key Constraints
- 2.27. Writing Seeders
- 2.28. Insert fake data using Model Factories
- 2.29. CRUD - Creating
- 2.30. CRUD - Reading
- 2.31. CRUD - Updating
- 2.32. CRUD - Deleting
- 2.33. CRUD Finishing
- 2.34. Using resource
- 2.35. Route
- 2.36. Using laravel form collective
- 2.37. Route model binding
- 2.38. Validation
- 2.39. Error handling
- 2.40. Soft Delete, Third Party Libraries and Notifications
- 2.41. Reading trashed data
- 2.42. Restore trashed data
- 2.43. Permanently delete trashed data
- 2.44. Laravel Debugbar
- 2.45. PDF
- 2.46. Excel
- 2.47. Email Notification
- 2.48. Database Notification
- 2.49. Middleware
- 2.50. Authentication (Default)
- 2.51. Email Verification
- 2.52. Authorization
- 2.53. Password Reset
- 2.54. Eloquent one to one relationship CRUD

- 2.55. Eloquent one to many relationship CRUD
- 2.56. Eloquent many to many relationship CRUD
- 2.57. Eloquent has one through & has many through
- 2.58. Eloquent one to one, one to many and many to many polymorphic relationship

