



PHP/MYSQL: LEVEL 1	2
OBJECTIVES	2
TARGET GROUP.....	2
TRAINING METHOD.....	3
COURSE DURATION.....	3
COURSE BREAKDOWN	3
PHP/MYSQL : LEVEL2	6
OBJECTIVES	6
TARGET GROUP.....	6
TRAINING METHOD.....	7
COURSE DURATION.....	7
PHP/MYSQL : LEVEL 3	11
OBJECTIVES	11
TARGET GROUP.....	11
TRAINING METHOD.....	11
COURSE DURATION.....	12
COURSE BREAKDOWN	12



PHP/MYSQL: LEVEL 1

OBJECTIVES

The DWIT Training - PHP/MYSQL course is targeted for beginners who want to:

- Learn how to think and write meaningful piece of code in PHP/MYSQL.
- Learn how to read PHP/MYSQL code that has been written by somebody else.
- Learn how to map literary description of a problem (requirement) to an application/library coded in PHP/MYSQL. In summary, this course teaches how to program using PHP/MYSQL programming language.

This is a core basic level course that is essential for anyone who have no prior programming experience but wish to be a professional PHP/MYSQL engineer in future

TARGET GROUP

Anyone who has some basic knowledge about programming and wants to learn to write applications in PHP/MYSQL for any purpose e.g. curiosity, hobby, to complete an academic project, to work towards a career as PHP/MYSQL programmer, to help in project management, etc.

Prerequisites

- Basic knowledge about programming, bits/bytes, procedures, classes, computer architecture, etc. If you just have a theoretical knowledge that is perfectly okay but you should have strong convictions on what programming is, and what you hope to achieve from this class.
- Willing and eager to spend at least 10-20 hours (varying from student-to-student) per week outside of the training class to read/write codes in PHP/MYSQL (self-study and practice).
- There are no prior educational level requirement for this course. Anyone from 10+2 student to someone who is doing her PHD in Genetic Engineering is welcome to take this course.
- If you are only interested in theory and have no interest/patience in spending at least 10 hours every week throughout the duration of the course, then this course is clearly not for you.
- If you have absolutely no idea about programming or do not see yourself doing programming in the next six -odd months, then this class may not be for you!



TRAINING METHOD

The course is spread over 40 hours that consists of lecture and lab work. There will be approximately 10 hours of lectures and 30 hours of hands-on lab work.

- Lab exercises are mandatory, have a fixed deadline, and are graded. The course puts heavy emphasis on lab exercises because software programming can only be learnt well by explicitly putting into practice the principles that have been taught (i.e. in simpler terms – by doing lots and lots of coding). Late submission (past the deadline) of exercises incur some penalty from total points.
- Instructors may provide relevant lecture/lab notes to students as (and when) necessary in the form of printed handouts and or via emails.
- Instructors may provide supplementary code snippets to students via email or in lab class to support the theory and or lab material that is being taught.
- At the end of the course, students may have to give an exam (which will be optional), that will test their knowledge on the material covered during the course. This exam may be practical and/or theoretical and is mandatory for any student wishing to join a higher level.
- Students are graded on the basis of attendance, lab exercises and exam in the increasing order of importance.

In summary, the only effective way to learn programming is to write lots of code. So in order to really make this training productive, students are encouraged to spend as much time as necessary to complete the lab exercises on time. As part of the course, students will spend at least 30 hours in the lab but specially if you are new to programming or are coming from a non-computer-science background, it is recommended that you spend at least 10-20 hours per week outside of the class on your own to practice coding in PHP/MYSQL.

COURSE DURATION

- 30 hours
- Classes
-Morning/Evening

COURSE BREAKDOWN

- **UNDERSTANDING THE PHP/ MYSQL BASICS**
 - How PHP works
 - The PHP.ini file
 - PHP tags
 - PHP Statements and Whitespaces
 - Comments
 - PHP functions



- **VARIABLES**
 - Variable types
 - Variables Names
 - Constant

- **FLOW CONTROLS**
 - If statements
 - For loop
 - While loop
 - Case statement

- **REUSING CODE AND WRITING FUNCTIONS**
 - Including files and Writing functions
 - Require
 - Require_once
 - Include
 - User functions
 - Defining and calling functions

- **INTRODUCTION TO DATABASE**
 - Basic difference between database and file system
 - Introduction of MySQL
 - Connection mechanism using PHP and MYSQL
 - Create table, select, insert, update and delete command
 - Making form with JQuery validation and inserting in table
 - Displaying all rows of table and disable in HTML table format.
 - Edit records using HTML form.
 - Delete records
 - Single-Dimensional Arrays
 - Multidimensional Arrays
 - Casting Arrays
 - Associative Arrays
 - Accessing Arrays
 - Getting the size of an array
 - Looping through an array
 - Looping through an associative array
 - Examining array
 - Joining arrays
 - Sorting arrays
 - Working EXAMPLES (Lab)



Labs

- Lab assignments will focus on the practice and mastery of contents covered in the lectures; and introduce critical and fundamental problem solving techniques to the students.

DISCLAIMER

Please note that Deerwalk Institute of Technology reserves the right to change the course syllabus of DWIT Training - PHP/MYSQL – Level 1 course at any time without prior notification.



PHP/MYSQL : LEVEL2

OBJECTIVES

The DWIT Training - PHP/MYSQL – Level 2 course is targeted for trainees:

- Who have had some prior beginner level hands-on programming experience in PHP/MYSQL programming language.
- Who have programming experience in some other programming language (e.g. PHP/MYSQL, Obj-C, PHP, C, C++, etc.) and want to learn PHP/MYSQL .

TARGET GROUP

- High school and university students (undergraduate, graduate, etc.) who want to do coursework (e.g. project, etc.) in PHP/MYSQL.
- Someone who has experience in some other programming language (e.g. C/C++, PHP, Perl, etc.), but has never done programming in ANDROID.
- Someone who is already working as a professional VB.NET developer and wants to switch to ANDROID.
- Someone who did her undergraduate in Economics, has been working in Media sector since graduation, and also working as a professional freelance PHP developer.
- Electrical/Electronic undergraduates in their 3rd semester who want to beef up their software skills prior to graduation.

Prerequisites

- Successfully complete the entrance test with score of at least 40% (for trainees directly applying to this level).
- Successfully complete the DWIT Training - PHP/MYSQL – Level 1 course (not applicable to trainees directly applying to this level).
- Successfully complete the interview.
 - o Willing and eager to spend at least 10-20 hours (varying from student-to-student) per week outside of the training class to read/write codes in PHP/MYSQL (self-study and practice).



TRAINING METHOD

The course is spread over 40 hours that consists of approximately 15 hours of lecture and 25 hours of hands-on lab work.

- Lab exercises are mandatory, have a fixed deadline, and are graded. The course puts heavy emphasis on lab exercises because software programming can only be learnt well by explicitly putting into practice the principles that have been taught (i.e. in simpler terms – by doing lots and lots of coding). Late submission (past the deadline) of exercises incur some penalty from total points.
- Instructors may provide relevant lecture/lab notes to students as (and when) necessary in the form of printed handouts and or via emails.
- Instructors may provide supplementary code snippets to students via email or in lab class to support the theory and or lab material that is being taught.
- At the end of the course, students may have to give an exam (which will be optional), that will test their knowledge on the material covered during the course. This exam may be practical and/or theoretical and is mandatory for any student wishing to join a higher level.
- Students are graded on the basis of attendance, lab exercises and exam in the increasing order of importance.

In summary, the only effective way to learn programming is to write lots of code. So in order to really make this training productive, students are encouraged to spend as much time as necessary to complete the lab exercises on time. As part of the course, students will spend at least 30 hours in the lab but specially if you are new to programming or are coming from a non-computer-science background, it is recommended that you spend at least 10-20 hours per week outside of the class on your own to practice coding in PHP/MYSQL.

COURSE DURATION

- 40 hours
- Classes
 - Morning/Evening



- **FUNCTIONS**

- What is a function
- Defining a function
- Returning value from function
- User-defined functions
- Variable scope
- Accessing variable with the global statement
- Function calls with the static statement
- Setting default values for arguments
- Passing arguments to a function by value
- Passing arguments to a function by reference
- Testing for function existence
- Working EXAMPLE (Lab)

- **WORKING WITH THE FILE SYSTEM**

- Creating and deleting a file
- Reading and writing text files
- Working with directories in PHP
- Checking for existence of file
- Determining file size
- Opening a file for writing, reading, or appending
- Writing Data to the file
- Reading characters
- Working Example

- **COMPLEX FORM PROCESSING**

- Super global variables
- The server array
- A script to acquire user input
- Importing user input
- Accessing user input
- Combine HTML and PHP code
- Using hidden fields
- Redirecting the user
- File upload and scripts
- PHP mail functions
- Form Security method and process.
- Working EXAMPLE (Lab)

- **WORKING WITH REGULAR EXPRESSIONS**



- The basic regular expressions
- Matching patterns
- Finding matches
- Replace patterns
- Working EXAMPLE (Lab)

- **CLASSES AND OBJECTS**

- Objects oriented programming
- Define a class
- An Object
- Creating an object
- Object properties
- Object methods
- Object constructors and destructors
- Class constants
- Class inheritance
- Abstract classes and methods
- Object serialization
- Checking for class and method existence
- Exceptions
- Iterators
- Summary
- Working EXAMPLE (Lab)

- **COOKIES**

- The anatomy of a cookie
- Setting a cookie with PHP
- Deleting a cookie
- Creating session cookie
- Working with the query string
- Creating query string
- Working EXAMPLES (Lab)

- **SESSION**

- What is session
- Starting a session
- Working with session variables
- Destroying session
- Passing session Ids
- Encoding and decoding session variables
- Working EXAMPLE (Lab)

- **SIMPLE JQUERY AJAX MODULE**

- Posting the form using Ajax I.e. without refreshing page.



- Changing content using Ajax.
- Form Validation
- Captcha Validation in FORMS (To prevent junk message from form)
- Working EXAMPLES (Lab)

Labs

Lab assignments will focus on the practice and mastery of contents covered in the lectures; and introduce critical and fundamental problem solving techniques to the students.

DISCLAIMER

Please note that Deerwalk Institute of Technology reserves the right to change the course syllabus of DWIT Training - PHP/MYSQL – Level 2 course at any time without prior notification.



PHP/MYSQL : LEVEL 3

OBJECTIVES

This course builds on the foundation laid by DWIT Training - PHP/MYSQL – Level 3 to prepare trainees for a career as PHP/MYSQL software engineer.

TARGET GROUP

Prerequisites

- Successfully completed the DWIT Training - PHP/MYSQL – Level 3 or obtained at least 40% score on the entrance exam.
- The latter case applies for new students that are directly attempting this training.
- Successfully complete the interview.
- Willing and eager to spend at least 10-20 hours (varying from student-to-student) per week outside of the training class to read/write codes in PHP/MYSQL (self-study and practice).
- Please note that this is a lab intensive course where the students will be expected to work on lab exercises for approximately half the duration of the session.

TRAINING METHOD

- The course is spread over 40 hours that consists of approximately 20 hours of lecture and 20 hours of lab work.
- Lab exercises are mandatory, have a fixed deadline, and are graded. The course puts heavy emphasis on lab exercises because software programming can only be learnt well by explicitly putting into practice the principles that have been taught (i.e. in simpler terms – by doing lots and lots of coding). Late submission (past the deadline) of exercises incur some penalty from total points.
- Instructors may provide relevant lecture/lab notes to students as (and when) necessary in the form of printed handouts and or via emails.
- Instructors may provide supplementary code snippets to students via email or in lab class to support the theory and or lab material that is being taught.



- At the end of the course, students may have to give an exam (which will be optional), that will test their knowledge on the material covered during the course. This exam may be practical and/or theoretical and is mandatory for any student wishing to join a higher level.
- Students are graded on the basis of attendance, lab exercises and exam in the increasing order of importance.

COURSE DURATION

- 30 hours
- Classes
-Morning/Evening

COURSE BREAKDOWN

- **INTRODUCTION TO LARAVEL**
 - Concept of Composer
 - Installing Composer
 - Installing Laravel using composer
 - Configuring Laravel
 - Project Structure
 - Setting up Development and Debugging Tools
- **ROUTING**
 - Basic Routing
 - Named Routing
 - Grouped Routing
 - Routing Parameters
 - Handling Invalid Routes
 - Route filtering with Middleware
- **CONTROLLERS**
 - Introduction
 - Basic Controllers
 - Restful Resource Controllers
 - Controller Routing
 - Controller Middleware
- **REQUEST**
 - Handling Request Info.
 - Request Inputs
 - Form Inputs



- Old Inputs
- Cookies
- Files

- **DATABASE**

- Introduction
- Migration
- Schema Building
- Migration Structure
- Writing Migration
- Running Migration
- Seeding
- Database Configuration
- Running Raw SQL queries
- Model
- Eloquent ORM
- Defining Model
- Retrieving Model
- Insert & Update Model
- Deleting Model
- Eloquent Relationship
- One to One
- One to Many
- Many to Many

- **VIEW**

- Passing data to view
- Sharing data to all views
- Blade Template Engine
- Introduction
- Template Inheritance
- Displaying Data
- Implement Control Structures

- **LARAVEL FORMS & HTML COMPONENT**

- Installation
- Generating form
- CSRF Protection
- Form Elements
- Labels
- Text, Text Area, Password & Hidden Fields
- Checkboxes and Radio Buttons
- File Input
- Number Input



- Data Input
- Drop-Down List
- Buttons
- Custom Form Macros

Labs

Lab assignments will focus on the practice and mastery of contents covered in the lectures; and introduce critical and fundamental problem solving techniques to the students.

DISCLAIMER

Please note that Deerwalk Institute of Technology reserves the right to change the course syllabus of DWIT Training - PHP/MYSQL – Level 2 course at any time without prior notification.