

## PHP Lectureflow

The below given flow should be followed by each faculty while taking lectures. If the faculty decides to change the flow - he/she will need to first take permission from the Training coordinator at the HO (Ahmedabad office)

| <b>Module 12)PHP - HTML, CSS, SQLite</b>  | <b>10</b> |
|---|-----------|
| <ul style="list-style-type: none"> <li>• HTML-(1 session)Basic HTML like What is Tags attribute property Form Tag HTML-HTML5 tags</li> <li>• CSS (1) • Selectors • Margin-Padding-display property • Create login-registration page</li> <li>• Bootstrap (2) Grid • Rows-Cols • Forms-Cards</li> <li>• SQL-(MySQLi)(4)</li> <li>• SQL Queries</li> <li>• Normalization</li> <li>• Joins</li> <li>• Indexer</li> <li>• Practical Examples 1) MySQL data types 2)Create Multiple Tables and perform the queries using Select, Update, Delete, Insert Where, Like, Group By, Having, Limit, Offset, Sub Query and Or, Not, In 3) Function and procedure 4) Trigger 5) Views 6) Primary and auto increment</li> </ul> |           |
| <b>Module 13) PHP - Core PHP</b>  | <b>10</b> |

- HTTP Protocol
- PHP syntax
- PHP variable
- Super Global Variables
- Practical Example: 1) Create Multiple Tables and perform the queries using Select, Update, Delete, Insert Where, Like, Group By, Having, Limit, Offset, Sub Query And , Or, Not, In
- Conditions Events and flows
- If condition
- If else if
- Practical Example : 1) Make Simple Calculator 2) Find the current day and if it is Sunday then print happy Sunday
- Practical Example: 1) Restaurant food Category Program using Switch case Display Which Category and dish user selected. 2) Using Ternary display age if it is greater than 18 3) Display Name Of the color which user selected
- Do While Loop
- For each Loop
- For loop
- Practical Example : 1) Create a script that displays 1-2-3-4-5-6-7-8-9-10 on one line. 2) Create a script using a for loop to add all the integers between 0 and 30 and display the total. 3) Write a PHP script using nested for loop that creates a chess board 4) All Patterns
- Array Function
- Date-time function
- Header function
- Include require
- Practical Example: 1) Perform all the function. 2) In user define function: 3) Create calculator 4) Find factorial using recursion 5) Reverse string without function 6) Download file on button click
- PHP array
- PHP expression
- PHP function
- PHP operations
- String Function
- PHP String
- Practical Example: 1) Display the value of the array 2) To Find the number of odd and even element in array. 3) Create associative array for user detail and display the user detail. 4) Shifted all the zero in bottom up of the array

- OOPs Concepts
- Class
- Object
- Extends
- Overloading
- Abstraction interface
- Constructor
- Destructor
- Magic Methods
- Scope Resolution
- Traits
- Visibility
- Type hinting
- Final keyword
- Email security function
- File handling
- Handling emails
- MVC Architecture
- Practical Example: Implementation of all the OOPS Concept
- Connection with MySQL database
- SQL injection
- Practical : 1) Take Project Definition 2) Exception handling with try catch for database connection and all queries. 3) Server- side Validation while Registration using Regular expressions. 4) Send mail while registration
- Session and cookies
- File upload
- PHP with MVC Architecture
- Insert, Update, Delete MVC
- Practical Example: 1) Apply session cookies concepts 2) File uploading and downloading 3) Bootstrap data-table Pagination
- Java script
- JQuery
- JQuery Events
- JQuery Validation
- Practical Example: 1)Alert, Prompt, Confirm 2)Create calculator 3)Create slideshow 4)Validation using JavaScript 5)Hide show using jQuery
- PHP and Ajax
- AJAX (crud)
- Regular Expression
- Practical Example :1) Event Examples (Onblur,onkeypress) 2)Regular expression 3) Create dependent drop down Pagination using ajax jQuery in MVC Project Insert Files using Ajax

## **Module 15) PHP - Web services, API, Extensions - Industry**

**4**

- Payment Gateway Integration
- -> Create API With Header -> API with Image Uploading
- Practical Example: Payment Gateway Implement on MVC Project
- -> SOAP and REST AP, Create API for insert, update and delete
- Product Catalog
- Shopping Cart
- Web Services
- Practical Example: Create Web Services for MVC Project
- Integration of API in Project
- RESTful principles, such as resource identification, statelessness, and uniform interfaces, is important for designing and consuming RESTful APIs.
- OpenWeatherMap API: This API provides weather data for various locations worldwide. You can retrieve current weather conditions, forecasts, and historical weather data.
- Google Maps Geocoding API: This API allows you to convert addresses into geographic coordinates (latitude and longitude) and vice versa. You can use it to retrieve location data, calculate distances between points, and display maps.
- GitHub API: GitHub provides an API that enables you to interact with repositories, issues, pull requests, and more. You can perform actions like retrieving repository information, creating issues, and accessing user data.
- Twitter API: Twitter offers an API that allows you to integrate Twitter functionality into your applications. You can fetch tweets, post tweets, retrieve user information, and perform searches.
- REST Countries API: This API provides information about countries, including details like population, languages spoken, currencies, and more. You can retrieve country-specific data and use it for various applications.
- SendGrid provides an API for sending transactional and marketing emails. You can integrate it into your applications to send emails, manage templates, and track delivery statistics.
- Social authentication (For eg; Login with Google, Login with Facebook...etc)
- Email sending APIs (For eg; Mailchimp, Mailgun...etc)
- SMS sending APIs (For eg; Twilio)
- Normal payments (For eg; Paypal, Stripe)
- - Google Maps API

|  |          |
|--|----------|
| <b>Module - 17) PHP - Introduction of Laravel PHP Framework - Industry</b> | <b>7</b> |
|--|----------|

|   |          |
|---|----------|
| <ul style="list-style-type: none"> <li>• Installing Laravel</li> <li>• Artisan CLI ( command-line interface )</li> <li>• Laravel Directory Structure</li> <li>• Configuring a new Laravel project</li> <li>• Artisan command to generate boilerplate code for a controller</li> <li>• Basics of laravel:----&gt; Routing</li> <li>• Controller</li> <li>• Middleware</li> <li>• Request</li> <li>• Response</li> <li>• HTML Template to Laravel Blade Template</li> <li>• HTML Template to Laravel Blade Template:-----&gt; Template inheritance</li> <li>• Master layout</li> <li>• Extending the master layout</li> <li>• Displaying variables</li> <li>• Blade conditional statements</li> <li>• Blade Loops</li> <li>• Executing PHP functions in blade</li> <li>• Displaying Your Views</li> <li>• Creating and using a basic views</li> <li>• Loading a view into another view/nested views</li> <li>• Adding assets</li> <li>• Integrating with Bootstrap</li> </ul> |          |
| <b>Module - 19) PHP - Laravel - Using Forms and Gathering Input - Industry</b>  | <b>8</b> |
| <ul style="list-style-type: none"> <li>• Creating contact us form</li> <li>• Validating user input</li> <li>• Sending email</li> <li>• Creating a file uploader</li> <li>• Validating a file uploader</li> <li>• Creating a custom error message</li> <li>• Laravel Collective:----&gt; Installation</li> <li>• Opening A Form</li> <li>• Form Model Binding</li> <li>• Form controls</li> <li>• Custom macros</li> </ul>   |          |
| <b>Module - 20) PHP - Laravel - Using Controllers and Routes for URLs and APIs - Industry</b>   | <b>5</b> |

- Introduction
- Creating a basic controller
- Creating a route using a closure
- Making the controller RESTful
- Using route groups
- Building a RESTful API with routes
- Rest API with Passport Authentication
- Event
- Mail
- Queue
- Task Scheduling
- Eloquent ORM and Models:----> Naming conventions
- Table name and primary keys
- Timestamps
- Eloquent ORM INSERT, READ, UPDATE, DELETE
- Relationship
- Using models in controllers
- Displaying data from models in views
- Creating and Using Composer Packages :----> Downloading and installing packages
- Using a basic Shopping Cart package in our project
- socialite
- Using Ajax and jQuery:---> Getting data from another page
- Setting up a controller to return JSON data
- Creating a JQuery, AJAX comment system
- Security & Session:---> Using Security
- Sessions Effective

## **Module - 21) PHP - Laravel - Migrations - Industry**

**2**

- Introduction
- Requirements for running migrations
- Artisan migration command
- Migration structure
- How to create a table using a migration
- Laravel migration rollback
- Database Seeding
- Migrations for our project database
- Pagination