PHP Training Program (2 Weeks)

Week 1 – Foundations

Day 1 – PHP Basics

Why Learn: Understand PHP syntax, variables, data types, and basic operators to build dynamic websites.

Tasks:

- Install XAMPP/LAMP stack.
- Write a PHP script for system info (PHP version, server details).
- Build a simple contact form (GET/POST handling).

Day 2 – Control Structures + Functions

Why Learn: Control flow and reusable functions are the building blocks of PHP apps.

Tasks:

- Implement loops, conditionals, and switch cases.
- Create functions with default/optional parameters.
- Write a utility library (string and array helpers).

Day 3 – Working with Forms + Sessions

Why Learn: Forms and sessions enable dynamic, stateful web applications.

Tasks:

- Create a registration form with validation.
- Store/retrieve data in PHP sessions.
- Implement flash messages (success/error).

Day 4 – PHP + MySQL Integration

Why Learn: Databases are essential for persistent data storage.

Tasks:

- Connect PHP to MySQL (mysqli/PDO).
- Build CRUD operations for a users table.
- Handle SQL errors gracefully.

Day 5 – File Handling + Uploads

Why Learn: Real-world apps require file management and media uploads.

Tasks:

- Read/write text files.
- Upload images with validation (size, type).
- Store file metadata in the database.

Day 6 – Authentication (Part 1)

Why Learn: Apps need secure login and signup.

Tasks:

- Build signup & login system with password hashing.
- Use sessions to manage login state.

Day 7 – Authentication (Part 2) + Cookies

Why Learn: Enhance auth with cookies and role-based access.

Tasks:

- Implement role-based authorization (admin/user).
- Add 'Remember Me' with cookies.
- Restrict routes based on roles.

Week 2 – Advanced Features

Day 8 – OOP in PHP

Why Learn: Object-Oriented PHP allows reusable, maintainable code.

Tasks:

- Create classes, objects, and inheritance.
- Implement a simple User model with methods.
- Use namespaces and autoloading.

Day 9 – Error Handling + Logging

Why Learn: Improve debugging and stability of PHP apps.

Tasks:

- Use try...catch for exceptions.
- Create a custom error handler.
- Log errors into files.

Day 10 – Security Best Practices

Why Learn: Protect apps from vulnerabilities (XSS, CSRF, SQL Injection).

Tasks:

- Implement prepared statements in MySQL.
- Escape user inputs.
- Add CSRF tokens in forms.

Day 11 – Pagination + Filtering

Why Learn: Manage large datasets efficiently.

Tasks:

- Implement pagination for posts.
- Add filtering and sorting options.

Day 12 – Project Structure + Git

Why Learn: Maintain clean architecture and collaborate with teams.

Tasks:

- Refactor into MVC structure (basic).
- Learn Git branching & PR workflow.

Day 13 – Documentation + Testing

Why Learn: Docs and tests ensure maintainability and reliability.

Tasks:

- Write PHPDoc comments.
- Create basic PHPUnit tests.
- Document API endpoints (if any).

Day 14 – Deployment + Capstone Project

Why Learn: Deploy and showcase real-world skills.

Tasks:

- Deploy app to shared hosting/VPS (cPanel or Apache/Nginx).
- Capstone: Build a Blog/Events Platform with auth, CRUD, file upload, search & deployment.