

PHP + MySQL Web Development



Web basics & environment setup

- How the web works (client, server, HTTP)
- Installing XAMPP / WAMP / MAMP
- File structure & localhost
- First PHP file: Hello World
- PHP tags, comments, echo & print, var_dump

PHP core syntax

- Variables, data types & constants
- Operators (arithmetic, comparison, logical)
- Type casting
- String functions (Example: strlen, str_replace, substr)
- Number functions (Example: round, abs, rand)

Control flow & loops

- if / elseif / else & switch / match
- for, while, do-while loops
- foreach loop for arrays
- break & continue
- Nested loops & loop patterns

MINI PROJECT 1

Grade Calculator

A PHP page that takes marks via a form and outputs grades, percentage, and pass/fail status.

Functions & scope

- Defining and calling functions
- Parameters, default values & return types
- Variable scope: local vs global
- Recursive functions
- Anonymous functions & arrow functions

Arrays

- Indexed & associative arrays
- Multidimensional arrays
- Array functions (Example: sort, array_map, array_filter)
- Spread operator & unpacking
- Searching with in_array & array_search

HTML forms & PHP

- \$_GET and \$_POST superglobals
- Form validation (required, type checks)
- Sanitizing input (htmlspecialchars, filter_var)
- File upload handling (\$_FILES)
- Redirect after POST pattern

MINI PROJECT 2

Student Registration Form

Multi-field registration form with validation, sanitization, and display of submitted data.

File handling

- Reading files (fopen, fread, file_get_contents)
- Writing & appending to files (fwrite, file_put_contents)

- Checking file existence (file_exists, is_file)
- File upload: save, rename & validate type/size
- Directory operations (mkdir, scandir, unlink)

Regular expressions

- Regex syntax: patterns, quantifiers, anchors
- preg_match – test if a pattern matches
- preg_match_all – find all matches
- preg_replace – search and replace
- Practical uses: email, phone, password validation

PHP & JSON

- What is JSON and when to use it
- json_encode – PHP array to JSON string
- json_decode – JSON string to PHP object/array
- Reading JSON files & storing config data
- Building a simple REST-style JSON response

MINI PROJECT 3

Contact Form with File Log

A contact form that validates inputs with regex, saves submissions to a JSON log file, and lets admin download the log

OOP fundamentals

- Classes, objects & instantiation
- Properties and methods
- Access modifiers: public, private, protected
- Constructors & destructors
- \$this keyword & method chaining

Inheritance & interfaces

- Extending classes (extends)
- Overriding methods & parent::
- Abstract classes & abstract methods
- Interfaces & implements
- Traits — reusing code across classes

OOP in practice

- Static properties & methods
- Magic methods (__toString, __get, __set)
- Namespaces & autoloading (spl_autoload)
- Exception handling (try / catch / finally)
- Organising code into classes (Database, User, etc.)

MINI PROJECT 4

OOP Student Management

Rebuild the earlier student form using classes: a Student class, a Validator class, and a FileLogger class.

MySQL fundamentals & CRUD

- Database design: tables, keys, relationships
- CREATE, ALTER, DROP table statements
- INSERT, SELECT, UPDATE, DELETE with WHERE
- ORDER BY, GROUP BY, HAVING, LIMIT
- Prepared statements with MySQLi & PDO

MySQL JOINS & relationships

- INNER JOIN — matching rows in both tables
- LEFT JOIN & RIGHT JOIN — include unmatched rows
- Self JOIN — joining a table to itself
- Multiple table joins (3+ tables)
- Practical: fetch users with their orders & products

Advanced MySQL

- Aggregate functions (COUNT, SUM, AVG, MIN, MAX)
- Subqueries — queries inside queries
- Views — virtual tables for complex queries
- Indexes for query performance
- Foreign key constraints & ON DELETE CASCADE

PHASE PROJECT 1

Library Management System

CRUD app with relational tables (books, authors, members) using JOINS to display data across tables

Sessions & cookies

- Starting and destroying sessions
- \$_SESSION for login state
- Login / logout / register flow
- Cookies (setcookie, \$_COOKIE, expiry)
- Flash messages & role-based access (admin/user)

Security essentials

- CSRF protection (token-based)
- Password hashing with password_hash()
- Secure file uploads (type, size, rename)

Sending email with PHP

- PHP mail() function basics
- Setting up PHPMailer via Composer
- Sending HTML emails with attachments
- Email templates (welcome, password reset)
- Using SMTP (Gmail / Mailtrap for testing)

PHASE PROJECT 2

Secure User Auth System

Full login/register system with hashed passwords, session management, CSRF tokens, and welcome email on registration via PHPMailer.

PROJECT

Web Application

Students will develop a dynamic web application with authentication, database, security, validations, file handling, and admin features.