Week 1: PHP Fundamentals I

Course: PHP & Laravel Backend Engineering

Instructor: AQTris Tech | GeekInk Innovations

Learning Objectives

By the end of this week, students should be able to: • Understand what PHP is and how it works • Write basic PHP syntax correctly • Use variables and data types • Apply operators in expressions • Control program flow using conditionals and loops • Build a simple result-checking application

Introduction to PHP

PHP (Hypertext Preprocessor) is a server-side scripting language designed for web development. It runs on the server and generates HTML output sent to the browser. PHP is widely used for backend development because it integrates easily with databases and web servers.

Basic PHP Syntax

• PHP code is written inside • Statements end with semicolons • PHP is case-sensitive for variables Example:

Variables and Data Types

Variables store data values and start with the $ sign. Common data types include: • String • Integer • Float • Boolean • Array Example: $name = "John"; $age = 20;

Operators

Operators are used to perform operations. Types include: • Arithmetic (+, -, *, /) • Comparison (==, !=, >, <) • Logical (&&, ||, !) Example: $total = $a + $b;

Conditional Statements

Conditional statements control decision-making. Common statements: • if • if...else • switch Example: if ($score >= 50) { echo "Pass"; } else { echo "Fail"; }

Loops

Loops execute code repeatedly. Types include: • for loop • while loop • foreach loop Example: for ($i = 1; $i <= 5; $i++) { echo $i; }

Mini Project: Student Result Checker

Students will build a PHP program that: • Accepts student name and score • Checks pass or fail • Displays result dynamically This project reinforces variables, conditionals, and output.

Summary

This week introduces the foundation of PHP programming. Mastery of these basics is essential before moving to forms, databases, and Laravel.