PHP & Laravel Backend Engineering

Week 1 – PHP Fundamentals I

Program: PHP & Laravel Backend Engineering
Week: 1
Level: Beginner
Duration: 3 Days

Learning Objectives

By the end of this week, students will be able to:

- ▯ Understand what PHP is and how it works
- ▯ Write basic PHP code with proper syntax
- ▯ Use variables and different data types
- ▯ Work with various operators
- ▯ Control program flow with conditions and loops
- ▯ Build a simple PHP project

Day 1 – Introduction to PHP

What is PHP?

PHP means **Hypertext Preprocessor**
It is a **server-side scripting language**

- ▯ PHP runs on the server
- ▯ It sends HTML output to the browser
- ⚙ Used for backend development
- ▯ Works well with databases like MySQL

Basic PHP Syntax

```
 <?php
// This is a single-line comment

echo "Hello World!";

/*
  This is a
  multi-line comment
*/
?>
```

Variables and Data Types

```
 $name = "John";     // String
$age = 20;           // Integer
$price = 10.5;       // Float
$isStudent = true;   // Boolean
```

Rules for variables:

- Start with `$`
- Must start with a letter or underscore
- Can only contain letters, numbers, and underscores
- Case-sensitive

Day 2 – Operators and Control Flow

Operators

Operator	Description	Example
`+`	Addition	`$sum = 5 + 3;`
`-`	Subtraction	`$diff = 10 - 4;`
`*`	Multiplication	`$product = 6 * 2;`
`/`	Division	`$quotient = 15 / 3;`
`%`	Modulus	`$remainder = 10 % 3;`

Conditional Statements

```
 // If-else statement
if ($age >= 18) {
    echo "You are an adult";
} else {
    echo "You are a minor";
}

// Switch statement
switch ($day) {
    case "Monday":
        echo "Start of work week";
        break;
    case "Friday":
        echo "Weekend is near!";
        break;
    default:
        echo "Regular day";
}
```

Loops

```
 // For loop
for ($i = 1; $i <= 5; $i++) {
    echo "Number: $i<br>";
}

// While loop
$count = 1;
while ($count <= 3) {
    echo "Count: $count<br>";
    $count++;
}

// Foreach loop (for arrays)
$colors = ["Red", "Green", "Blue"];
foreach ($colors as $color) {
    echo "$color<br>";
}
```

Day 3 – Student Result Checker Project

Project Requirements

Build a PHP program that:

1. Takes student name and score as input
2. Calculates the grade based on the score

3. Displays the result in a clean format

Grading System:

- 90-100: A+
- 80-89: A
- 70-79: B
- 60-69: C
- 50-59: D
- Below 50: F

Sample Code Structure

```
<?php
$name = "Ahmed";
$score = 78;
$grade = '';

if ($score >= 90) $grade = 'A+';
elseif ($score >= 80) $grade = 'A';
elseif ($score >= 70) $grade = 'B';
elseif ($score >= 60) $grade = 'C';
elseif ($score >= 50) $grade = 'D';
else $grade = 'F';

echo "<h2>Student Result</h2>";
echo "<p>Name: $name</p>";
echo "<p>Score: $score</p>";
echo "<p>Grade: $grade</p>";
?>
```

Practice Exercises

1. Create a simple calculator that can add, subtract, multiply, and divide two numbers
2. Write a PHP script that prints all even numbers from 1 to 20
3. Create a program that checks if a number is prime
4. Build a simple form that collects user's name and email, then displays a welcome message

Resources

- □ PHP Official Documentation
- □ PHP: The Right Way
- □ W3Schools PHP Tutorial

Tips for Success

- □ Practice coding every day
- □ Test your code with different inputs
- □ Take notes on important concepts
- □ Collaborate with classmates
- □ Use online resources when stuck