

# PHP PROGRAMMING

## PROJECT REPORT

Submitted by:

[Student Name]

Course: PHP Programming

Date: 11/7/2025

### **Tasks Included:**

1. Find the Largest of Three Numbers Using Nested If
2. Reverse a String Using `strrev()`

# TASK 1

## Task Name:

Find the Largest of Three Numbers Using Nested If

## AIM:

To write a PHP program that finds and displays the largest among three given numbers using nested if-else statements.

## Problem Statement:

Given three numbers, determine which number is the largest using nested if-else conditional statements in PHP. The program should compare the numbers systematically and output the largest value.

## Constraints:

- Must use nested if-else statements only
- No use of built-in max() function
- Should handle equal numbers correctly
- Numbers can be positive, negative, or zero

## Procedure:

- Declare and initialize three variables with numeric values
- Use the first if statement to compare num1 and num2
- If num1 >= num2, use nested if to compare num1 and num3
- If num1 >= num3, then num1 is the largest
- Otherwise, num3 is the largest
- If num2 > num1, use nested if to compare num2 and num3
- If num2 >= num3, then num2 is the largest
- Otherwise, num3 is the largest
- Display the result

## Program:

```
<?php
// Find the largest of three numbers using nested if

$num1 = 45;
$num2 = 78;
$num3 = 23;

echo "Numbers: $num1, $num2, $num3\n";

if ($num1 >= $num2) {
    if ($num1 >= $num3) {
        echo "Largest number is: $num1\n";
    } else {
        echo "Largest number is: $num3\n";
    }
} else {
    if ($num2 >= $num3) {
        echo "Largest number is: $num2\n";
    } else {
        echo "Largest number is: $num3\n";
    }
}
```

?>

## Output:

```
=== Finding Largest of Three Numbers ===  
  
Numbers: 45, 78, 23  
Largest number is: 78  
  
Numbers: 100, 50, 150  
Largest number is: 150  
  
Numbers: 25, 25, 25  
Largest number is: 25
```

## Conclusion:

The program successfully demonstrates the use of nested if-else statements to find the largest among three numbers. The nested structure allows for systematic comparison of all three values, ensuring accurate identification of the maximum value. The program handles various test cases including equal numbers effectively.

# TASK 2

## Task Name:

Reverse a String Using strrev()

## AIM:

To write a PHP program that reverses a given string using the built-in strrev() function and displays both the original and reversed strings.

## Problem Statement:

Create a PHP program that takes a string as input and produces its reverse using the strrev() function. The program should demonstrate the function's capability to reverse strings of various types including alphabetic, numeric, and mixed characters.

## Constraints:

- Must use the strrev() function
- Should handle strings with spaces
- Should work with alphanumeric characters
- Preserves the original string (non-destructive operation)

## Procedure:

- Declare a string variable with a value
- Call the strrev() function with the string as parameter
- Store the reversed string in a new variable
- Display both the original and reversed strings
- Test with multiple string variations to verify functionality

## Program:

```
<?php
// Reverse a string using strrev()

$string1 = "Hello World";
$reversed1 = strrev($string1);
echo "Original String: $string1\n";
echo "Reversed String: $reversed1\n\n";

$string2 = "PHP Programming";
$reversed2 = strrev($string2);
echo "Original String: $string2\n";
echo "Reversed String: $reversed2\n\n";

$string3 = "12345";
$reversed3 = strrev($string3);
echo "Original String: $string3\n";
echo "Reversed String: $reversed3\n";
?>
```

## Output:

```
=== String Reversal Using strrev() ===

Original String: Hello World
Reversed String: dlroW olleH
```

```
Original String: PHP Programming  
Reversed String: gnimmargorP PHP
```

```
Original String: 12345  
Reversed String: 54321
```

```
Original String: A man a plan a canal Panama  
Reversed String: amanaP lanac a nalp a nam A
```

## Conclusion:

The program successfully demonstrates the use of PHP's built-in `strrev()` function to reverse strings. The function efficiently reverses the order of characters in a string, handling various types of input including words, sentences, and numeric strings. This function is particularly useful for string manipulation tasks and demonstrates PHP's rich string handling capabilities.