Lecture #2 PHP

A Server-Side Scripting Language

The comment in PHP

- In PHP, a comment is a piece of text within the code that is not executed or interpreted by the PHP engine. Comments are used to add explanatory notes, documentation, or temporarily disable certain parts of the code. They help programmers understand the code and make it more readable. PHP supports two types of comments:

Example

```
<?php
// This is a single-line comment
$variable = 10; // Assigning the value 10 to the variable
?>
```

```
<?php
/*
This is a multi-line comment.
It can span multiple lines.
*/
$result = 5 * 8; // Calculate the result
?>
```

Multidimensional Arrays:

A multidimensional array is an array of arrays. It allows you to organize data in a tabular form with rows and columns.

```
// Multidimensional array representing a 2x3 matrix
$matrix = array(
   array(1, 2, 3),
   array(4, 5, 6)
);

// Accessing elements
echo $matrix[0][1]; // Output: 2
```

Associative Arrays:

- An associative array uses named keys instead of numeric indices. This allows you to associate values with specific names.

- // Associative array representing a person
- $person = array(
-     'name' => 'John Doe',
-     'age' => 25,
-     'city' => 'Exampleville'
-);

- // Accessing elements
- echo $person['name']; // Output: John Doe

Arrays with Different Data Types:

PHP arrays can store elements of different data types within the same array.

```
// Array with different data types
$mixedArray = array(
   'name' => 'Alice',
   'age' => 30,
   'isStudent' => false,
   'grades' => array(90, 85, 92)
);

// Accessing elements
echo $mixedArray['grades'][1]; // Output: 85
```

Assign by Value:

- $originalVar = 10;
- $newVar = $originalVar; // Assigning by value

- $newVar = 20;

- echo $originalVar; // Output: 10
- echo $newVar;   // Output: 20

Assign by Reference:

- $originalArray = [1, 2, 3];
- $referencedArray = &$originalArray; // Assigning by reference

- $referencedArray[] = 4;

- print_r($originalArray); // Output: Array([0] => 1, [1] => 2, [2] => 3, [3] => 4)
- print_r($referencedArray); // Output: Array([0] => 1, [1] => 2, [2] => 3, [3] => 4)