Unit-III
Lecture-I
Array

An array is a special variable, which can hold more than one value at a time.
An array can store multiple elements of similar data type under a single variable thereby saving us the effort of creating a different variable for every data. The arrays are helpful to create a list of elements of similar types, which can be accesses using their index or key.

An array is created using an **array()** function in PHP.

There are basically three types of arrays in PHP:

- **Indexed or Numeric Arrays:** An array with a numeric index where values are stored linearly.
- **Associative Arrays:** An array with a string index where instead of linear storage, each value can be assigned a specific key.
- **Multidimensional Arrays:** An array which contains single or multiple array within it and can be accessed via multiple indices.

Indexed or Numeric Arrays:

These types of arrays can be used to store any type of elements, but an index is always a number. By default, the index starts at zero.

These arrays can be created in two different ways as shown below:

```php
<?php

// One way to create an indexed array
$fruits = array("Apple", "Mango", "Pineapple", "Cherry", "Banana");

// Accessing the elements directly
echo "Accessing the 1st array elements directly:\n";
echo $fruits[2], "\n";
echo $fruits[0], "\n";
echo $fruits[4], "\n";
echo "</br></br>";

// Second way to create an indexed array
$fruits[0] = "Apple";
$fruits[1] = "Mango";
$fruits[2] = "Pineapple";
$fruits[3] = "Cherry";
$fruits[4] = "Banana";

// Accessing the elements directly
echo "Accessing the 2nd array elements directly:\n";
echo $fruits[2], "\n";
echo $fruits[0], "\n";
echo $fruits[4], "\n";

?>
```

Associative Arrays:

These types of arrays are similar to the indexed arrays but instead of linear storage, every value can be assigned with a user defined key of string type.

```php
<?php
// define associatve array first method
$data = array(
 'username' => 'adeven',
 'password' => 'secret',
 'host' => '192.168.0.1'
);

echo $data['username'];
echo "</br> </br>";

// define associative array second method
$data['username'] = 'adeven';
$data['password'] = 'secret';
$data['host'] = '192.168.0.1';

echo $data['username'];
echo ", your password is " .$data['password'];
?>
```

Note: To assign all the values of an array in a single element, set a key for each value and link the two using the **=>** connection and remember to separate each key-value pair with commas.

Multidimensional Arrays:

Multi-dimensional arrays are such arrays that store another array at each index instead of a single element. In other words, we can define multi-dimensional arrays as an array of arrays.

Every element in this array can be an array and they can also hold other sub-arrays within. Arrays or sub-arrays in multidimensional arrays can be accessed using multiple dimensions.

```php
<?php

// Defining a multidimensional array
$favorites = array(
    array(
        "name" => "Rachel",
        "mob" => "9458748795",
        "email" => "rachel@gmail.com",
    ),
    array(
        "name" => "Chandler",
        "mob" => "9874525487",
        "email" => "chandler@gmail.com",
    ),
    array(
        "name" => "Ross",
        "mob" => "7098155789",
        "email" => "ross@gmail.com",
    )
);

// Accessing elements
echo "Rachel email-id is: " . $favorites[0]["email"];
echo "</br></br>";
echo "Ross mobile number is: " . $favorites[2]["mob"];

?>
```