

What is PHP Array?

A **PHP Array** is a data structure that allows you to store multiple values in a single variable. It can hold different data types, including numbers, strings, or even other arrays, and is a versatile way to manage collections of related data.

Use Cases of PHP Arrays:

1. **Storing a List of Items:** Useful for storing multiple values like a list of products or names.
 2. **Mapping Key-Value Pairs:** Associative arrays can map keys to values, like storing user details.
 3. **Storing Data from a Database:** Arrays are often used to hold data retrieved from databases for easier manipulation.
 4. **Implementing Stack or Queue:** Arrays can be used to implement data structures like stacks or queues.
 5. **Passing Multiple Values to Functions:** Arrays can be passed to functions to handle multiple related values.
-

Indexed Arrays

You can create an indexed array using the `array()` function or the short array syntax `[]`.

```
$array1 = array("apple", "banana", "cherry");
$array2 = ["apple", "banana", "cherry"];

//Accessing Values
echo $array1[0];
echo $array2[0];

//For Loop Through an Indexed Array
for ($i = 0; $i < count($array1); $i++) {
    echo $array1[$i] . "<br>";
}

//ForEach Loop Through an Indexed Array
foreach ($array1 as $value) {
    echo $value . "<br>";
}
```

Associative arrays

- Associative arrays in PHP are arrays that use named keys instead of numeric indexes.
- This - makes them more expressive and useful when dealing with key-value pairs.
- Associative arrays can be defined using the `array()` function or the short array syntax `[]`.

```
// How to create
$person1 = array("first_name" => "John", "last_name" => "Doe");
$person2 = ["first_name" => "Jane", "last_name" => "Smith"];

// Accessing Values
echo $person1["first_name"];

//Looping Through an Associative Array
foreach ($person1 as $key => $value) {
    echo "$key : $value<br>";
}
```

Multidimensional arrays

Multidimensional arrays are arrays that contain other arrays as their elements. The contained arrays can be indexed or associative

```
//Declaring a Multidimensional Array
$array = array(
    array(1, 2, 3),
    array(4, 5, 6),
    array(7, 8, 9)
);

//Accessing Elements
echo $array[1][2];

//Associative Multidimensional Array
$users = array(
    "john" => array(
        "age" => 25,
        "email" => "john@example.com"
    ),
    "jane" => array(
        "age" => 30,
        "email" => "jane@example.com"
    )
);

echo $users["john"]["email"];

//Looping Through Multidimensional Arrays
foreach($users as $username => $details) {
    echo "Username: $username\n";
    foreach($details as $key => $value) {
        echo "$key: $value\n";
    }
}
```

#php