Arrays

Qué es un array

En PHP, un array es una estructura de datos que nos permite almacenar múltiples valores bajo un solo nombre. Cada valor en el array tiene una posición numérica asociada a ella, conocida como índice, que se utiliza para acceder y manipular los elementos almacenados.

Un array en PHP puede contener cualquier tipo de dato, incluyendo números, cadenas de texto, booleanos, otros arrays e incluso objetos. Esto lo convierte en una herramienta poderosa para almacenar y organizar datos de manera eficiente.

Propiedades y características

Índices numéricos: Por defecto, los arrays en PHP utilizan índices numéricos que comienzan en 0 y se incrementan en 1 para cada elemento adicional. Estos índices numéricos permiten acceder a elementos individuales del array de manera directa.

Índices asociativos: Además de los índices numéricos, PHP también permite utilizar índices asociativos, que son cadenas de texto o números utilizados como claves para acceder a los elementos del array. Los índices

asociativos brindan mayor flexibilidad al permitirnos asignar nombres descriptivos a los elementos almacenados.

Tamaño dinámico: A diferencia de otros lenguajes de programación que requieren especificar el tamaño de un array al crearlo, en PHP los arrays tienen un tamaño dinámico. Esto significa que podemos agregar o eliminar elementos del array en cualquier momento sin preocuparnos por el tamaño inicial.

Tipos de datos mixtos: Los arrays en PHP pueden contener diferentes tipos de datos en sus elementos. Esto nos permite almacenar una mezcla de números, cadenas, booleanos e incluso otros arrays dentro de un solo array.

Arrays multidimensionales: Un array en PHP también puede contener otros arrays como elementos, lo que da lugar a los arrays multidimensionales. Esto nos permite crear estructuras de datos complejas con múltiples niveles de profundidad.

Funciones y operaciones especializadas: PHP ofrece una variedad de funciones integradas para manipular y trabajar con arrays, como ordenar, buscar elementos, unir arrays, y más. Estas funciones simplifican la gestión y el procesamiento de datos almacenados en arrays.

Tipos de arrays y sintaxis

En PHP, hay varias formas de declarar un array.

Simples

Array vacio

```
$miArray = [];
$miArray = array();
```

Array vacío pero con longitud

```
$miArray = [10];
$miArray = array(10);
```

Array con valores

```
$miArray = [1, 2, 3, 4, 5];
$miArray = array(1, 2, 3, 4, 5);
```

Asociativo

```
$miArray = array(
    'clave1' => 'valor1',
    'clave2' => 'valor2',
    'clave3' => 'valor3'
);
```

Multidimensional

```
$miArray = array(
    array($v1, $v2, $v3),
    array($v4, $v5, $v6),
    array($v7, $v8, $v9)
);
```

Funciones predefinidas para trabajar con arrays

- count(): Obtener la cantidad de elementos en un array.

```
count($miArray)
```

- in_array(): Verificar si un valor existe en un array.

```
in_array($valorBuscado, $miArray)
```

- array_key_exists(): Verificar si una clave existe en un array.

```
array_key_exists($claveBuscada, $miArray)
```

- array_push(): Agregar elementos al final de un array.

```
array_push($miArray, 60, 70)
```

-array_pop(): Eliminar el último elemento de un array.

```
array_pop($miArray)
```

Ordenar

- sort(): Ordenar arrays numéricos de forma ascendente.

```
sort($miArray)
```

- rsort(): Ordenar arrays numéricos de forma descendente.

```
rsort($miArray)
```

- asort(): Ordenar arrays asociativos de forma ascendente según los valores.

```
asort($persona)
```

- arsort(): Ordenar arrays asociativos de forma descendente según los valores.

```
arsort($persona)
```

Unir y dividir

- array_merge(): Unir arrays.

```
$frutas = array('manzana', 'naranja', 'banana');
$verduras = array('zanahoria', 'espinaca', 'pepino');
$alimentos = array_merge($frutas, $verduras)
```

- array_slice(): Dividir arrays en subconjuntos.

```
$subconjunto = array_slice($alimentos, 1, 3)
```

Operaciones básicas con arrays

> ⚠️ **ADVERTENCIA** a partir de este punto no entraremos en explicar los arrays asociativos ya que se verán exclusivamente en una sección a parte SOON **EN EL HORNO** curso sobre arrays multidimensionales en PHP

Acceder a elementos

Para acceder a elementos de un array en PHP, utilizamos los índices numéricos o asociativos según el tipo de array que estemos usando.

Arrays Simples

```
$miArray = array(10, 20, 30, 40, 50);

echo $miArray[0]; // Imprime 10
echo $miArray[3]; // Imprime 40
```

Arrays Asociativos

```
$persona = array(
    'nombre' => 'Juan',
    'edad' => 30,
    'ciudad' => 'Madrid'
);

// Acceder a elementos usando índices asociativos
echo $persona['nombre']; // Imprime "Juan"
echo $persona['edad']; // Imprime 30
```

Agregar elementos

Arrays Simples

```
$miArray = array(10, 20, 30);

// Agregar un elemento al final del array
$miArray[] = 40;

// Agregar un elemento en una posición específica
$miArray[1] = 25;
```

Arrays Asociativos

```
$persona = array(
    'nombre' => 'Juan',
    'edad' => 30
);

// Agregar un elemento al array asociativo
$persona['ciudad'] = 'Madrid';
```

Eliminar elementos

Arrays Simples

```php
$miArray = array(10, 20, 30, 40, 50);

// Eliminar el último elemento del array
array_pop($miArray);

// Eliminar un elemento en una posición específica
unset($miArray[1]);
```

Arrays Asociativos

```php
$persona = array(
    'nombre' => 'Juan',
    'edad' => 30,
    'ciudad' => 'Madrid'
);

// Eliminar un elemento del array asociativo
unset($persona['ciudad']);
```

Arrays predefinidos

En PHP, existen algunos arrays predefinidos que proporcionan información útil sobre el entorno y los datos del programa.

- $_GET: Contiene los datos enviados a través de una petición HTTP GET.
- $_POST: Contiene los datos enviados a través de una petición HTTP POST.
- $_SESSION: Almacena variables de sesión, que persisten a lo largo de múltiples páginas durante una sesión del usuario.
- $_COOKIE: Contiene las cookies enviadas por el navegador del usuario.
- $_SERVER: Contiene información sobre el servidor y el entorno de ejecución.
- $_FILES: Contiene información sobre los archivos subidos al servidor a través de un formulario.
- $_ENV: Contiene las variables de entorno del sistema.