



Type Hinting

2º DAW - Desarrollo Web en Entorno Servidor

David Soler Talens



Que es Type Hinting?

- **PHP es un lenguaje débilmente tipado**
- **PHP aplica Inferencia de Tipos**
- **La *inferencia de tipos* es un proceso por el cual el intérprete determina el Tipo de una Variable sin una declaración explícita de su Tipo**
- **Este proceso puede fallar, por ejemplo**
 - "2" se convierte fácilmente en 2 -> "2" + 2 = 4
 - "palabra" fallaría al ser convertido en numérico -> "palabra" + 2 = error
- **Usaremos Type hinting para especificar los tipos y evitar el error anterior**
- **PHP también permite utilizar Type Hinting para las propiedades y métodos de la clase**

Parámetros de función



```
function my_function(type $param1, type param2, ...) {  
    // ...  
}
```

Tipo

Parámetro

```
function add(int $x, int $y)  
{  
    return $x + $y;  
}  
  
$result = add(1,2);  
echo $result; // 3
```

Si pasamos algo diferente a un entero se produce un **TypeError**, excepto si enviamos un "tipo compatible", como un float, dado que la conversión es posible. Podemos forzar al tipo exacto con "strict typing"

[Strict Typing en php.net](https://www.php.net/manual/es/strict-tying.php)

Tipos



Tipo	Descripción	Versión
Nombre de la Clase/Interfaz	El valor debe ser una instancia de la clase o de la interfaz.	
self	El valor debe ser una instanceof de la misma clase que aquella en la que se utiliza la declaración de tipo. Solamente se puede usar en clases.	
parent	El valor debe ser una instanceof del padre de la clase en la que se usa la declaración de tipo. Solo se puede usar en clases.	
array	El valor debe ser un array.	
callable	El valor debe ser un callable válido. No puede ser usado como una declaración de tipo de propiedad de clase.	
bool	El valor debe ser un valor booleano.	
float	El valor debe ser un número de coma flotante.	
int	El valor debe ser un número entero.	
string	El valor debe ser un string.	
iterable	El valor debe ser un array o una instanceof de la clase Traversable .	PHP 7.1.0
object	El valor debe ser un objeto.	PHP 7.2.0
mixed	El valor puede ser cualquier valor.	PHP 8.0.0

Valor de retorno de función



```
function my_function(type $param1, type $param2, ...) : type
{
    // ..
}
```

Tipo del valor retornado con "return"

```
function add(int $x, int $y): int
{
    return $x + $y;
}
```

```
echo add(10, 20);
```

El valor retornado es un entero, por lo que no se produce ningún mensaje de error.
Si la función no retorna nada, el tipo será "void"



Otros tipos (I)

- **Unión**: permite devolver un conjunto posible de dos o más tipos

```
function add($x, $y): int | float
{
    return $x * $y;
}

echo add(10, 20); // 200 (int)
echo add(1.5, 2.5); // 3.75 (float)
```

Otros tipos (II)



- **Nullable**: permite que un valor pueda ser de un determinado tipo o, como alternativa, ser nulo. Se especifica con el carácter "?"

```
function upper(?string $str): string
{
    return strtoupper($str);
}

$str = null;
echo upper($str);
```

Otros tipos (III)



- **Mixed**: El tipo mixto significa "uno de varios tipos". Es equivalente al siguiente tipo de unión

```
object|resource|array|string|int|float|bool|null
```


Type Hinting e include/require



Cuando se llama a una función definida en un archivo CON Strict Typing desde un archivo SIN Strict Typing mediante 'include/require' o cualquiera de sus variantes, PHP respetará la preferencia del llamante.

Esto significa que es el llamador el que decide si usar el modo estricto o no.