

Parámetros

Los parámetros son los datos que reciben las funciones y que utilizan para realizar las operaciones de esa función. Una función puede recibir cualquier número de parámetros, incluso ninguno.

Si la función que estamos construyendo no necesita recibir ningún parámetro, al declararla, simplemente indicamos los paréntesis vacíos en la cabecera. Por ejemplo en la siguiente función mostramos la fecha del día de hoy. Para ello nos apoyamos en otra función incluida en PHP: `date()`.

Ejercicio 1:

```
function fecha_hoy() {  
    echo date('d/m/Y');  
}
```

En el caso que queramos, o necesitemos, recibir parámetros para implementar una función, a la hora de definirla, en la cabecera, se definen los parámetros que va a recibir.

Así definimos una función llamada `f1` que recibe dos parámetros. Como se puede observar, no se tiene que definir el tipo de datos de cada parámetro. Es decir, la función necesitará que le enviemos dos datos, pero no le importará que sean de un tipo u otro.

Los parámetros tienen validez durante la ejecución de la función. Se dice que tienen un ámbito local a la función donde se están recibiendo. Cuando la función se termina, los parámetros dejan de existir.

Los parámetros se pasan por valor

El paso de parámetros en PHP se realiza por valor. "Por valor" es una manera típica de pasar parámetros en funciones, quiere decir que el cambio de un dato de un parámetro no actualiza el dato de la variable que se pasó a la función. Por ejemplo, cuando invocamos una función pasando una variable como parámetro, a pesar de que cambiemos el valor del parámetro dentro de la función, la variable original no se ve afectada por ese cambio.

Ejercicio 2: paso de parámetros por valor

```
function porvalor ($parametro1){  
    $parametro1="hola";  
    echo "<br>" . $parametro1; //imprime "hola"  
}  
  
$mivariable = "esto no cambia";  
porvalor ($mivariable);  
echo "<br>" . $mivariable; //imprime "esto no cambia"
```

Paso de parámetros por referencia

En contraposición al paso de parámetros por valor, está el paso de parámetros por referencia. En este último caso, el cambio del valor de un parámetro dentro de una función sí afecta al valor de la variable original.

Podemos pasar los parámetros por referencia si, en la declaración de la función, colocamos un "&" antes del parámetro.

Ejercicio 3: Pase de parámetros por referencia.

```
function porreferencia(&$cadena) {  
    $cadena = 'Si cambia';  
}  
  
$str = 'Esto es una cadena';  
porreferencia ($str);  
echo $str; // Imprime 'Si cambia'
```

Parámetros por defecto

Podemos definir valores por defecto para los parámetros. Los valores por defecto sirven para que los parámetros contengan un dato predefinido, con el que se inicializarán si no se le pasa ningún valor en la llamada de la función. Los valores por defecto se definen asignando un dato al parámetro al declararlo en la función.

Ejemplo de una función con parámetros por defecto:

```
function pordefecto ($parametro1="pepe";$parametro2=3)
```

Ejercicio 4: Realizar un script en PHP que contenga una función con valores de parámetros por defecto y ejemplificar el uso de la función con parámetros y sin parámetros.