

ÉRICA PALOMINO

TEMA 8

FUNCIONES

erica.palomino@escuelaartegranada.com

ESCUELAARTEGRANADA

INTRODUCCIÓN

Es una sección de código separada a la que se le ha dado un nombre.

Puede ser llamada(invocada) tantas veces como se quiera.

Dividen tareas para hacer más comprensible el script PHP

Pueden recibir / devolver valores.

INTRODUCCIÓN

Sintaxis:

```
function nombreFunción ( parámetros ){
```

```
    Sentencias;
```

```
}
```

INTRODUCCIÓN

Para llamar a una función sólo hay que escribir una sentencia que **contenga** el nombre de la función, seguida de los **parámetros**.

La ejecución salta a la línea en la que empieza la función.



PASO DE PARÁMETROS

PASO DE PARÁMETROS

PHP permite pasar los parámetros:

- **Por valor.** No se modifica el valor original de la variable.
- **Por referencia.** Se modifica el valor de la variable.
- **Por defecto.** Valor que se le da al parámetro en caso de que no se le pase de forma externa.

PASO DE PARÁMETROS POR VALOR

Es la opción por defecto en PHP

La función recibe **una copia del valor** de la variable.

No hay problema en modificarla.

```
function funcion1 ( $var1 ){
```

```
    Sentencias;
```

```
}
```

PASO DE PARÁMETROS POR REFERENCIA

En algunas ocasiones necesitaremos **modificar el valor** original del parámetro recibido.

PHP permite recibir la dirección de memoria de la variable original.

Sólo habrá que añadir un **&** **delante del parámetro** en la declaración de la función.

```
function funcion1 ( &$var1 ){  
    Sentencias;  
}
```


PASO DE PARÁMETROS POR DEFECTO

Son la forma en que PHP implementa los parámetros opcionales. Toman el valor predefinido, cuando al llamar la función, **no se les pasa un argumento**.

Para definirlos hay que escribir, **en la cabecera** de la función:

- Nombre del parámetro
- Operador de asignación (=)
- Valor por defecto

Los parámetros pasados por defecto deben ir **SIEMPRE** los últimos.



DEVOLUCIÓN DE VALORES

DEVOLUCION DE VALORES

Se pueden devolver **cualquier tipo de valor**.

SÓLO se puede devolver un valor

Para devolver **varios valores**, devolveremos un **array**.

Utilizaremos la palabra reservada **return**.

Cuando se encuentra un **return la función termina**.



ÁMBITO DE LAS VARIABLES

VARIABLES GLOBALES

Definidas fuera del cuerpo de una función.

Accesibles desde cualquier punto del código.

Para ello deben ser declaradas:

- Anteponiendo la palabra clave 'global'.
- Usando el array asociativo `$GLOBALS` (con el nombre de la variable como clave y su contenido como valor).

VARIABLES LOCALES

Definidas dentro del cuerpo de una función.

Accesibles **SÓLO** desde esa función.

Si se declara una variable 'local' dentro de una función, con el mismo nombre que una variable 'global', se trabajará con la variable 'local'


VARIABLES ESTÁTICAS

No confundir con las variables static de Java.

Son variables locales.

No pierden su valor entre las diferentes llamadas a la función que las definen.

Hay que anteponerle la palabra static.

The background features two large, decorative curved shapes. A blue shape is in the top right corner, and a yellow shape is in the bottom left corner. Both are thick, curved bands that do not form complete circles.

FUNCIONES CON NÚMERO VARIABLE DE PARÁMETROS

NÚMERO VARIABLE DE PARÁMETROS

PHP nos permite definir funciones en las que el número de parámetros no está fijado a priori.

No se necesita una sintaxis específica, pero su funcionamiento se basa en el uso de:

func_num_args()

- Devuelve el número de parámetros recibidos por la función

NÚMERO VARIABLE DE PARÁMETROS

`func_get_args()`

- Devuelve un array con los argumentos pasados a la función

`func_get_arg()`

- Devuelve un elemento de la lista de argumentos pasados.
- Los argumentos empiezan en la posición CERO
- Si se solicita una posición que no existe => *false*



**INCLUIR
FICHEROS**

INCLUIR FICHEROS

En PHP podemos crear un archivo .php que incluya una serie de funciones o variables que usamos frecuentemente.

Podremos incluir ese fichero en cualquier otro fichero utilizando la función **require_once**

Con **require_once** indicamos la ruta relativa del archivo que queremos incluir.