

# UT2. Inserción de código en páginas Web, el lenguaje de programación PHP

DAW - Desarrollo Web Entorno Servidor  
Fernando Galindo

1. Inserción de código
2. Introducción a PHP
3. Comentarios
4. Variables
5. Estructuras de control

# 1. Inserción de código

Los códigos PHP se encuentran dentro de las etiquetas `<?php y ?>`

Puede estar embebido dentro del código HTML

```
<?php
    echo '<?xml version="1.0" encoding="UTF-8"?>'
?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="es" lang="es">
<head> <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Título</title>
</head>
<body>
<?php
    echo "Código PHP";
?>
</body>
</html>
```

## 2. Introducción a PHP

### Las reglas básicas para escribir PHP

- Todas las líneas deben acabar con punto y coma ;
- Se puede agrupar el código utilizando llaves { y }
- Una línea se puede sangrar o partir (varias líneas) con el fin de que sea más legible, con la condición de no partir palabras o valores
- PHP obliga a ser estricto con las mayúsculas y minúsculas en algunos casos como las variables, pero no con las palabras reservadas

`WHILE`, `while` y `While` son lo mismo

`$var`, `$VAR` y `$vAr` son variables diferentes

## 3. Comentarios

### Comentarios

```
/*
```

```
Comentario varias líneas
```

```
*/
```

```
$a=5; // Comentario una línea
```

Los comentarios NO se muestran en el lado de cliente

## 4. Variables

### Variables

- Tiene que empezar con el símbolo \$. Ese símbolo es el que permite distinguir a una variable de otro elemento del lenguaje PHP.
- El segundo carácter puede ser \_ o una letra
- El tercer carácter y siguientes puede contener números, letras o \_
- No hay restricción de tamaño
- No puede contener espacios en blanco

Recomendación, los nombres de las variables sean significativos para el entorno

`$saldo`, `$saldo_inicial` y `$saldoFinal`

`$x123` es válido pero no es un buen nombre

## 4. Variables

### Números

```
$entero=5;
```

```
$octal=071; // Muestra 56 por pantalla, empieza 0
```

```
$hexa=0xA2BC; // Muestra 41660 por pantalla, empieza 0x
```

## 4. Variables

### Texto

```
$nombre="Fernando Galindo";
```

```
$frase = `Bienvenidos al módulo "DWES" de DAW`;
```

El texto **DWES** aparece entre comillas, la variable **empieza** por comillas simples



## 4. Variables

### Secuencias de escape

Son útiles si queremos añadir caracteres especiales en nuestro código

secuencia de escape	significado
\t	Tabulador
\n	Nueva línea
\f	Alimentación de página
\r	Retorno de carro
\"	Dobles comillas
\'	Comillas simples
\\	Barra inclinada (backslash)
\\$	Símbolo dólar

## 4. Variables

Si queremos utilizar secuencias de escape debemos utilizar comillas dobles

```
$frase = "Bienvenidos al módulo \"DWES\" de DAW";
```

**Bienvenidos al módulo "DWES" de DAW**

Las barras aparecen por pantalla

```
$frase = `Bienvenidos al módulo \"DWES\" de DAW`;
```

**Bienvenidos al módulo \"DWES\" de DAW**

## 4. Variables

Podemos incluir dentro del texto el valor de una variable

```
$dia = 270;
```

```
$frase = "Faltan $dia para el verano";
```

**Cuidado** con las comillas simples

```
$frase = 'Faltan $dia para el verano';
```

Faltan 270 para el verano  
Faltan \$dia para el verano

## 4. Variables

PHP es un lenguaje no tipado

```
$entero=5;
```

```
$real=3.2;
```

```
$resultado= $entero + $real; //Devuelve un real
```

```
$entero= $entero + $real; //$entero contiene un real
```

Casting de tipos ([documentación](#))

```
$resultado= $entero + (int) $real; // La suma es 5+3
```

```
//Se queda con la parte entera del número
```

## 4. Variables

Variables, declaración y asignación:

```
$a = 5;
```

```
$modulo = 'DWES';
```

```
$curso = 2;
```

No necesitamos especificar el tipo.

Variables reservadas:

<https://www.php.net/manual/es/reserved.variables.server.php>

## 5. Estructuras de control

Un código de PHP podemos encapsular con los símbolos { } y corresponde a un bloque de código

A mayores tenemos 2 tipos de estructuras

- Condicionales
- Repeticiones

## 5. Estructuras de control

### Condicionales (if & switch)

```
if ($a < $b)
    echo "$a es menor que $b";
elseif ($a > $b)
    echo "$a es mayor que $b";
else
    echo "a es igual a b";
```

```
switch ($a) {
    case 0:
        echo "a vale 0";
        break;
    case 1:
        echo "a vale 1";
        break;
    default:
        echo "a no vale 0 ni 1";
}
```

## 5. Estructuras de control

### Repeticiones (while, do & for)

```
$a = 1;
```

```
while ($a < 8)
```

```
    $a += 3;
```

```
echo $a; // el valor obtenido es 10
```

```
$a = 5;
```

```
do
```

```
    $a -= 3;
```

```
while ($a > 10);
```

```
echo $a; // el bucle se ejecuta una sola vez, con lo que el valor obtenido es 2
```

```
for ($a = 5; $a<10; $a+=2) {  
    echo $a."<br/>;  
}
```