



upcomillas *es*

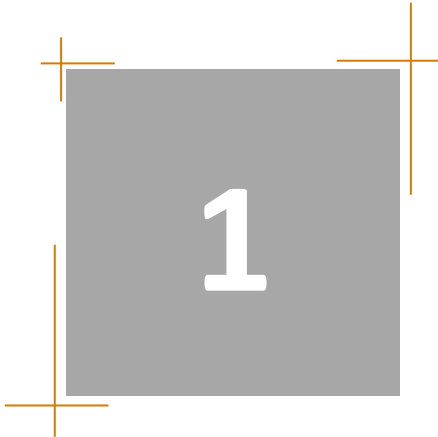
upcomillas *es*

4-Introducción a PHP

Curso: Desarrollo de aplicaciones web

Cristina Puente, Rafael Palacios

2010-2011



Introducción a PHP



¿Qué es PHP?

- Lenguaje de alto nivel que se ejecuta en el servidor
- Al ejecutarse en el servidor, todas las páginas van a poder ser vistas en cualquier ordenador, independientemente del navegador que tenga.
- Es gratuito
- Los ficheros PHP pueden contener texto, etiquetas HTML y código PHP o scripts
- Tienen como extensión de fichero ".php", ".php3", or ".phtml"



Variables y Constantes

- La forma de representar variables en php es anteponiendo el símbolo \$ a la palabra que utilizaremos como variable.
- Son meros contenedores de datos, no tienen tipo asociado, pero sí que debemos saber cuándo utilizar comillas a la hora de definirlas.
 - \$a=5;
 - \$b="7";
- Son sensibles a mayúsculas y minúsculas \$a != \$A
- Una misma variable puede contener un número y luego puede contener caracteres.

Variables y Constantes

```
<body>
  <?php
    $a = 1;
    $b = 3.34;
    $c = "Hola Mundo";
    echo $a, "<br>", $b, "<br>", $c;
  ?>
</body>
```

Impresión de variables

- Las variables se pueden imprimir mediante 'echo' o mediante 'printf' :

```
<html>
<head>
    <title>Ejemplo de PHP</title>
</head>
<body>
Parte de HTML normal.
<BR><BR>
<?php
    echo "Parte de PHP<br>";
?>
</body>
</html>
```

Impresión de variables

- printf permite formatear la salida
- La cadena de formateo indica cómo se han de representar los valores que posteriormente le indicaremos.
- La principal ventaja es que además de poder formatear los valores de salida, nos permite intercalar texto entre ellos.

```
<?php
    $var="texto";
    $num=3;
    printf("Puede fácilmente intercalar <b>%s</b> con
           números <b>%d</b> <br>", $var, $num);

    printf("<TABLE BORDER=1 CELLPADDING=20>");
    for ($i=0;$i<10;$i++)
    {
        printf("<tr><td>%10.d</td></tr>", $i);
    }
    printf("</table>");
?>
```

Operadores

- Similares a C y a Javascript

Operador	Nombre	Ejemplo	Descripción
+	Suma	5 + 6	Suma dos números
-	Resta	7 - 9	Resta dos números
*	Multiplicación	6 * 3	Multiplica dos números
/	División	4 / 8	Divide dos números
%	Módulo	7 % 2	Devuelve el resto de dividir ambos números, en este ejemplo el resultado es 1
++	Suma 1	\$a++	Suma 1 al contenido de una variable.
--	Resta 1	\$a--	Resta 1 al contenido de una variable.

Operadores

- Comparación:

Operador	Nombre	Ejemplo	Devuelve cierto cuando:
==	Igual	$\$a == \b	$\$a$ es igual $\$b$
!=	Distinto	$\$a != \b	$\$a$ es distinto $\$b$
<	Menor que	$\$a < \b	$\$a$ es menor que $\$b$
>	Mayor que	$\$a > \b	$\$a$ es mayor que $\$b$
<=	Menor o igual	$\$a <= \b	$\$a$ es menor o igual que $\$b$
>=	Mayor o igual	$\$a >= \b	$\$a$ es mayor o igual que $\$b$

Operadores

- Lógicos:

Operador	Nombre	Ejemplo	Devuelve cierto cuando:
&&	Y	<code>(7>2) && (2<4)</code>	Devuelve verdadero cuando ambas condiciones son verdaderas.
and	Y	<code>(7>2) and (2<4)</code>	Devuelve verdadero cuando ambas condiciones son verdaderas.
	O	<code>(7>2) (2<4)</code>	Devuelve verdadero cuando al menos una de las dos es verdadera.
or	O	<code>(7>2) or (2<4)</code>	Devuelve verdadero cuando al menos una de las dos es verdadera.
!	No	<code>! (7>2)</code>	Niega el valor de la expresión.

Bloques de Flujo If-else

- La sentencia if ejecuta una serie de instrucciones u otras dependiendo de la condición que le pongamos
- Si no hemos incluido una etiqueta else, continuará la ejecución del resto de instrucciones

```
<?php
    $a = 8;
    $b = 3;
    if ($a < $b)
    {
        echo "a es menor que b";
    }
    else
    {
        echo "a no es menor que b";
    }
?>
```

Bloques de Flujo Switch

- Con la sentencia switch podemos ejecutar unas u otras instrucciones dependiendo del valor de una variable

```
<?php
    $posicion = "arriba";
    switch($posicion) {
        case "arriba":    // Bloque 1
            echo "La variable contiene";
            echo " el valor arriba";
            break;
        case "abajo":    // Bloque 2
            echo "La variable contiene";
            echo " el valor abajo";
            break;
        default:        // Bloque 3
            echo "La variable contiene otro valor";
            echo " distinto de arriba y abajo";
    }
?>
```

Bucles For

- El primer parámetro del for, sirve para inicializar la variable del bucle
- El segundo parámetro indica la condición que se debe cumplir para que el bucle siga ejecutándose
- El tercer parámetro es una instrucción que se ejecuta al final de cada iteración y sirve para modificar el valor de la variable de iteración

```
<?php
for ($i=0;$i<10;$i++)
{
    echo "El valor de i es ", $i,"<br>";
}
?>
```

Bucles While & Do-While

- Nos permiten iterar conjuntos de instrucciones, es decir repetir la ejecución de un conjunto de instrucciones mientras se cumpla una condición.

```
<?php
    $i=0;
    while ($i<10)
    {
        echo "El valor de i es ", $i,"<br>";
        $i++;
    }
?>
```

Bucles Foreach

- Esta instrucción se ejecuta para recorrer matrices, lo que puede resultar muy útil cuando se recuperan datos de una base de datos
- No se necesita conocer el número de filas del array, ya que se para cuando finaliza (comienza procesando la primera fila y salta a la siguiente en cada iteración).

```
<?php
    foreach($array as $valor)
    {
        echo "<td>",$valor,"</td>";
    }
?>
```



Escuela Técnica Superior de Ingeniería ICAI

Alberto Aguilera 25
28015 Madrid
Tel +34 91 542 28 00
Fax + 34 91 542 31 76
lwww.icaí.upcomillas.es

www.upcomillas.es

