

Manual de PHP Básico - WebEstilo

Iniciación a PHP

- [Conceptos Básicos](#)
- [Nuestro primer PHP](#)
- [Variables](#)

Operadores en PHP

- [Aritméticos](#)
- [Comparación](#)
- [Lógicos](#)

Instrucciones en PHP

- [Condicionales](#)
- [Bucles](#)
- [Salida](#)
- [Cadenas](#)

Funciones

- [Funciones](#)
- [Librerías](#)

Ejemplos prácticos

- [Formateo de página](#)
- [Enlace externo](#)

1. Iniciación a PHP

1.1. Conceptos básicos

El lenguaje PHP es un lenguaje de programación de estilo clásico, con esto quiero decir que es un lenguaje de programación con variables, sentencias condicionales, bucles, funciones.... No es un lenguaje de marcas como podría ser HTML, XML o WML. Está mas cercano a JavaScript o a C, para aquellos que conocen estos lenguajes.

Pero a diferencia de Java o JavaScript que se ejecutan en el navegador, PHP se ejecuta en el servidor, por eso nos permite acceder a los recursos que tenga el servidor como por ejemplo podría ser una base de datos. El programa PHP es ejecutado en el servidor y el resultado enviado al navegador. El resultado es normalmente una página HTML pero igualmente podría ser una pagina WML.



Al ser PHP un lenguaje que se ejecuta en el servidor no es necesario que su navegador lo soporte, es independiente del navegador, pero sin embargo para que sus páginas PHP funcionen, el servidor donde están alojadas debe soportar PHP.

1.2. Nuestro primer PHP

La ventaja que tiene PHP sobre otros lenguajes de programación que se **ejecutan en el servidor** (como podrían ser los script CGI Perl), es que nos permite intercalar las sentencias PHP en las paginas HTML, es un concepto algo complicado de entender si no se ha visto nunca como funciona unas paginas PHP o ASP.

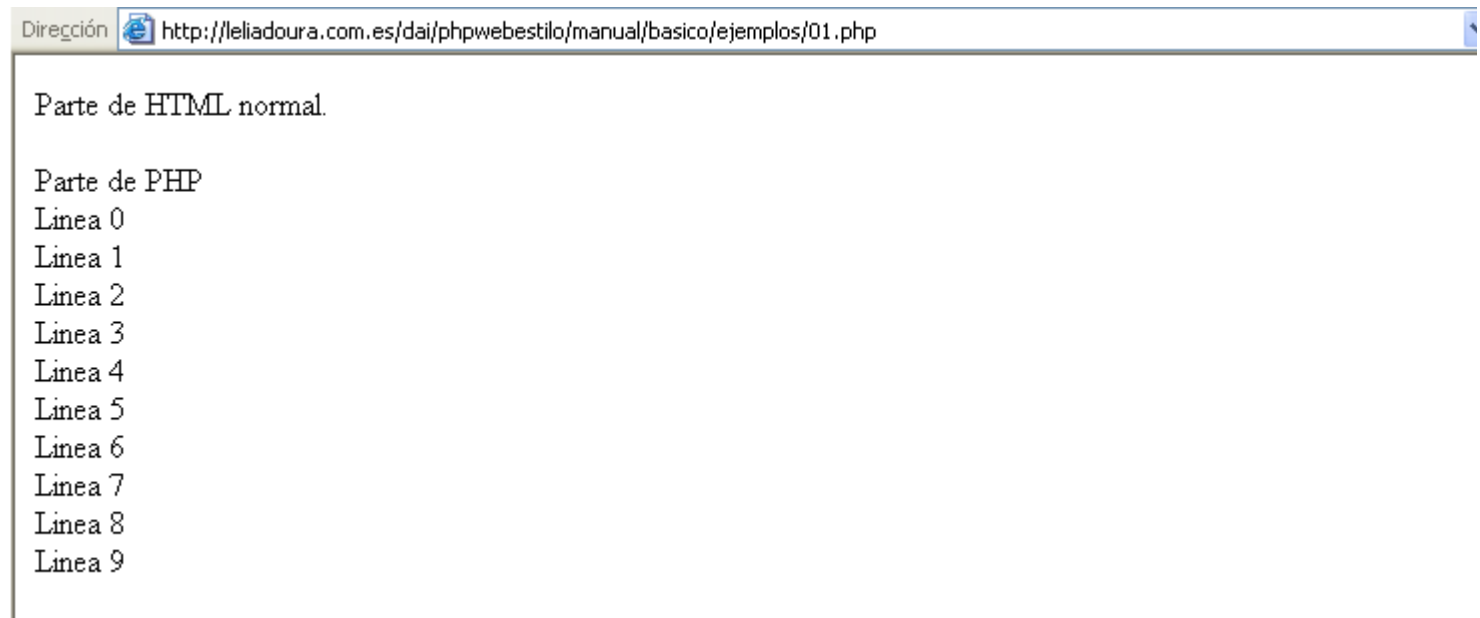
Vamos a ver un ejemplo sencillo para comprenderlo mejor. En azul está el código HTML y en rojo el código PHP. Seguiremos este criterio durante todo el manual.

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 01 WebEstilo Basico </title>
  </head>
  <body>
    Parte de HTML normal.
    <br><br>
    <?php
      echo "Parte de PHP <br>";
      for ($i=0;$i<10;$i++)
      {
        echo "Linea ".$i."<br>";
      }
    ?>
  </body>
</html>
```

[Ejecutar código](#)

Vemos la ejecución



El código PHP ejecutado tiene dos partes: la primera imprime "Parte de PHP" y la segunda es un bucle que se ejecuta 10 veces de 0 a 9, por cada vez que se ejecuta se escribe una línea, la variable $\$i$ contiene el número de línea que se está escribiendo.

No importa si no entiende muy bien el programa este ejemplo solo es para ilustrar como se **intercala el código HTML y el código PHP**.

1.3. Variables

Una variable es un contenedor de información, en el que podemos meter números enteros, números decimales, caracteres... el contenido de las variables se puede leer y se puede cambiar durante la ejecución de una página PHP.

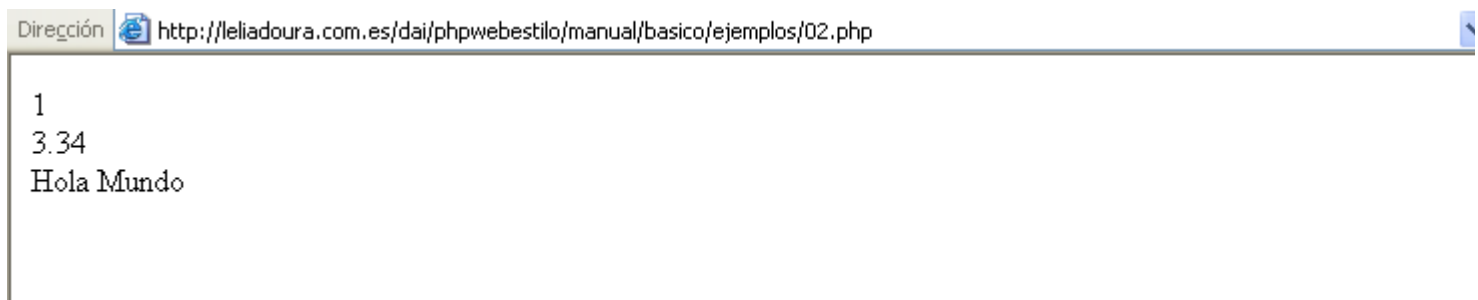
En PHP todas las variables comienzan con el símbolo del dólar \$ y no es necesario definir una variable antes de usarla. Tampoco tienen tipos, es decir que una misma variable puede contener un número y luego puede contener caracteres.

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 02 WebEstilo Basico </title>
  </head>
  <body>
    <?php
      $a=1;
      $b=3.34;
      $c="Hola Mundo";
      echo $a, "<br>", $b, "<br>", $c;
      // la linea anterior genera lo mismo que
      // las 5 lineas siguientes
      /* echo $a;
      echo "<br>";
      echo $b;
      echo "<br>";
      echo $c; */
    ?>
  </body>
</html>
```

[Ejecutar código](#)

Vemos la ejecución



En este ejemplo hemos definido tres variables, \$a, \$b y \$c y con la instrucción echo hemos impreso el valor que contenían, insertando un salto de línea entre ellas.

Existen 2 tipos de variables, las variables locales que solo pueden ser usadas dentro de funciones y las variables globales que tienen su ámbito de uso fuera de las funciones, podemos acceder a una variable global desde una función con la instrucción global nombre_variable;

2. Operadores en PHP

2.1. Aritméticos

Los operadores de PHP son muy parecidos a los de C y JavaScript, si usted conoce estos lenguajes le resultaran familiares y fáciles de reconocer.

Estos son los operadores que se pueden aplicar a las variables y constantes numéricas.

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 caso el resultado es 1
++	Suma 1	\$a++	Suma 1 al contenido de una variable.
--	Resta 1	\$a--	Resta 1 al contenido de una variable.

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 03 WebEstilo Basico </title>
  </head>

  <body>
    <?php
      $a=8;
      $b=3;
      echo $a+$b, "<br>";
      echo $a-$b, "<br>";
      echo $a*$b, "<br>";
      echo $a/$b, "<br>";
      $a++;
      echo $a, "<br>";
      $b++;
      echo $b, "<br>";
    ?>
  </body>
</html>
```

[Ejecutar código](#)

Vemos la ejecución



```
Dirección http://leliadoura.com.es/dai/phpwebestilo/manual/basico/ejemplos/03.php

11
5
24
2.6666666666667
9
4
```

2.2. Comparación

Los operadores de comparación son usados para comparar valores y así poder tomar decisiones.

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

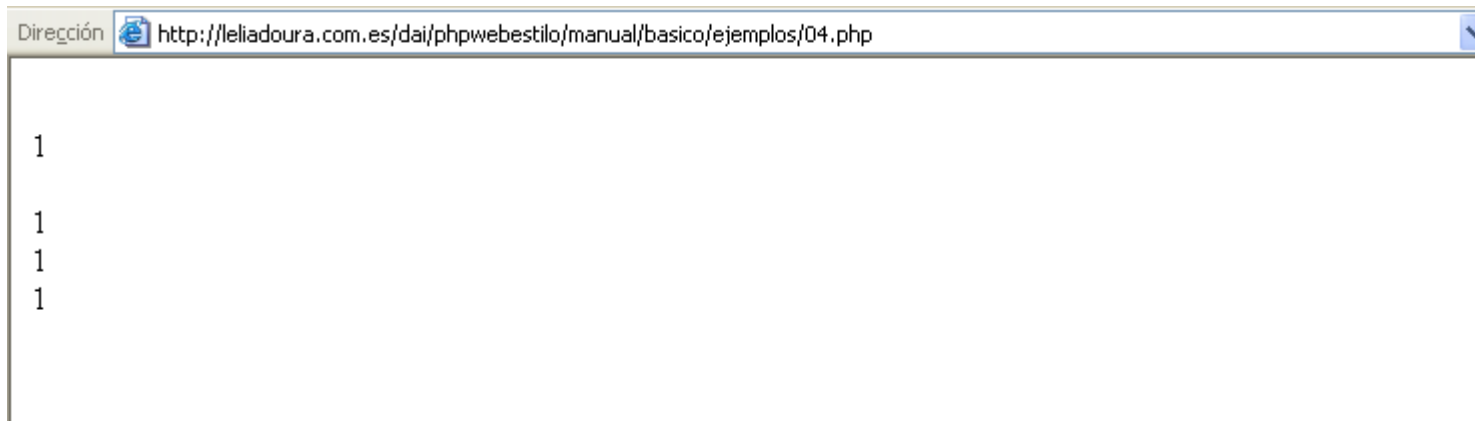
Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 04 WebEstilo Basico </title>
  </head>

  <body>
    <?php
      $a=8;
      $b=3;
      $c=3;
      echo $a==$b, "<br>";
      echo $a!=$b, "<br>";
      echo $a<$b, "<br>";
      echo $a>$b, "<br>";
      echo $a>=$c, "<br>";
      echo $b<=$c, "<br>";
    ?>
  </body>
</html>
```

[Ejecutar código](#)

Vemos la ejecución



2.3. Lógicos

Los operadores lógicos son usados para evaluar varias comparaciones, combinando los posibles valores de estas.

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

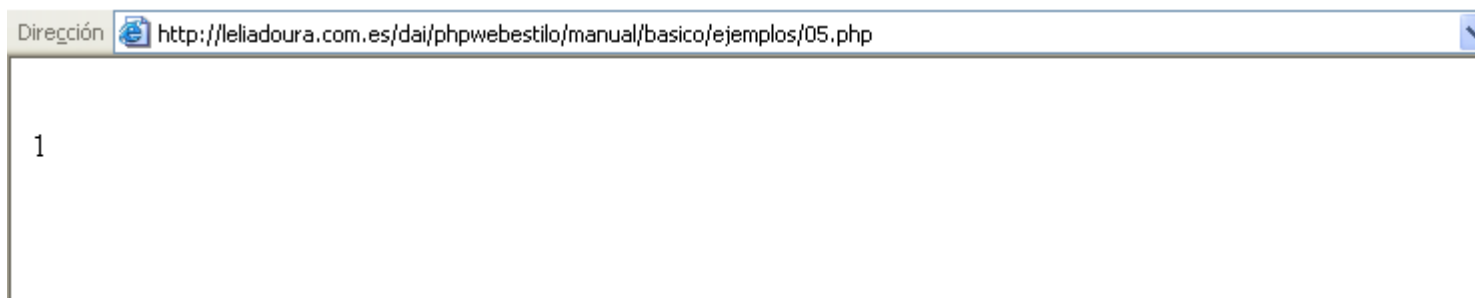
Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 05 WebEstilo Basico </title>
  </head>

  <body>
    <?php
      $a=8;
      $b=3;
      $c=3;
      echo ($a==$b)&&($c>$b) , "<br>";
      echo ($a==$b) || ($b==$c) , "<br>";
      echo !($b<=$c) , "<br>";
    ?>
  </body>
</html>
```

[Ejecutar código](#)

Vemos la ejecución



3. Instrucciones en PHP

3.1. Condicionales

Las sentencias condicionales nos permiten ejecutar o no unas ciertas instrucciones dependiendo del resultado de evaluar una condición. Las más frecuentes son la instrucción if y la instrucción switch.

Sentencia if ... else

```
<?php
  if (condición)
  {
    Sentencias a ejecutar cuando la
    condición es cierta.
  }
```

```

else
{
    Sentencias a ejecutar cuando la
    condición es falsa.
}
?>

```

La sentencia if ejecuta una serie de instrucciones u otras dependiendo de la condición que le pongamos. Probablemente sea la instrucción más importante en cualquier lenguaje de programación.

Código fuente

```

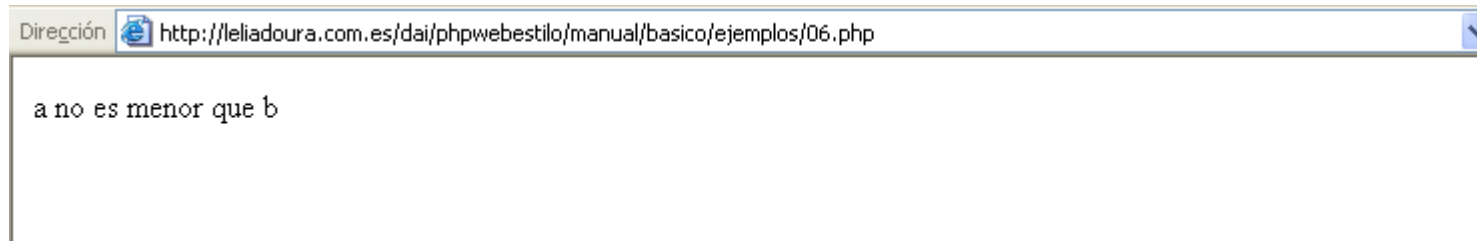
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 06 WebEstilo Basico </title>
  </head>

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

```

[Ejecutar código](#)

Vemos la ejecución



En este ejemplo la condición no es verdadera por lo que se ejecuta la parte de código correspondiente al else.

Sentencia switch ... case

Código fuente

```

<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 07 WebEstilo Basico </title>
  </head>
  <body>
    <?php
      $posicion="arriba";
      switch($posicion)
      {
        case "arriba": // bloque 1
          echo "La variable contiene";
          echo " el valor arriba";
          break;

```

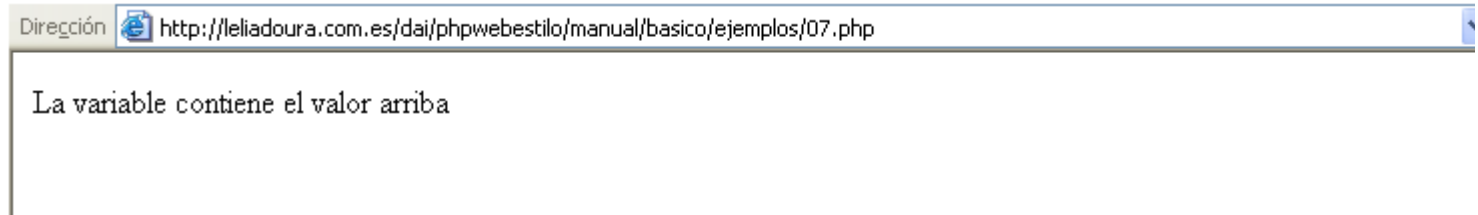
```

        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";
    }
    ?>
</body>
</html>

```

[Ejecutar código](#)

Vemos la ejecución



Con la sentencia switch podemos ejecutar unas u otras instrucciones dependiendo del valor de una variable, en el ejemplo anterior, dependiendo del valor de la variable \$posicion se ejecuta el bloque 1 cuando el valor es "arriba", el bloque 2 cuando el valor es "abajo" y el bloque 3 si no es ninguno de los valores anteriores.

3.2. Bucles

Los bucles nos permiten **iterar conjuntos de instrucciones**, es decir repetir la ejecución de un conjunto de instrucciones mientras se cumpla una condición.

Sentencia while

```

<?php
while (condición)
{
    intrucciones a ejecutar.
}
?>

```

Mientras la condición sea cierta se reiterará la ejecución de las instrucciones que están dentro del while.

Código fuente

```

<!-- Manual de PHP de WebEstilo.com -->
<html>
<head>
<title> Ejemplo 08 WebEstilo Basico </title>
</head>
<body>
Inicio <br>
<?php
    $i=0;
    while ($i<10)
    {
        echo "El valor de i es ",$i,"<br>";
        $i++;
    }
?>
Final <br>
</body>
</html>

```


[Ejecutar código](#)

Vemos la ejecución



```
Dirección http://leliadoura.com.es/dai/phpwebestilo/manual/basico/ejemplos/08.php

Inicio
El valor de i es 0
El valor de i es 1
El valor de i es 2
El valor de i es 3
El valor de i es 4
El valor de i es 5
El valor de i es 6
El valor de i es 7
El valor de i es 8
El valor de i es 9
Final
```

En el siguiente ejemplo, el valor de `$i` al comienzo es 0, durante la ejecución del bucle, se va sumando 1 al valor de `$i` de manera que cuando `$i` vale 10 ya no se cumple la condición y se termina la ejecución del bucle.

Sentencia for

```
<?php
for (inicial ; condición ; ejecutar en iteración)
{
    instrucciones a ejecutar.
}
?>
```

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 09 WebEstilo Basico </title>
  </head>
  <body>
    Inicio <br>
    <?php
      for ($i=0;$i<10;$i++)
        echo "El valor de i es ",$i,"<br>";
    ?>
    Final <br>
  </body>
</html>
```

[Ejecutar código](#)

Vemos la ejecución



```
Dirección http://leliadoura.com.es/dai/phpwebestilo/manual/basico/ejemplos/09.php

Inicio
El valor de i es 0
El valor de i es 1
El valor de i es 2
El valor de i es 3
El valor de i es 4
El valor de i es 5
El valor de i es 6
El valor de i es 7
El valor de i es 8
El valor de i es 9
Final
```

La instrucción `for` es la instrucción de bucles más completa. En una sola instrucción nos permite controlar todo el funcionamiento del bucle.

El primer parámetro del `for`, es ejecutado la primera vez y sirve para inicializar la variable del bucle, el segundo parámetro indica la condición que se debe cumplir para que el bucle siga ejecutándose y 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.

3.3. Salida

Hasta ahora hemos usado la instrucción `echo` para realizar salida a pantalla, esta instrucción es bastante limitada ya que no nos permite formatear la salida. En esta página veremos la instrucción `printf` que nos da mucha más potencia.

Sentencia `printf`

```
<?php
printf(cadena formato, variable1, variable2...);
?>
```

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.

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>

<head>
  <title> Ejemplo 10 WebEstilo Basico </title>
</head>
```

```

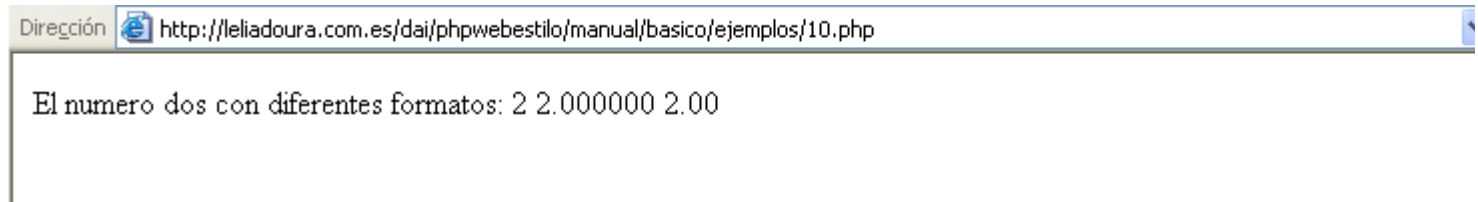
<body>
  <?php
    printf ("El numero dos con diferentes formatos: %d %f %.2f",2,2,2);
  ?>
</body>

</html>

```

[Ejecutar codigo](#)

Vemos la ejecución



La cadena de formato puede incluir una serie de caracteres especiales que indican como formatear las variables que se incluyen en la instrucción.

Elemento	Tipo de variable
%s	Cadena de caracteres.
%d	Número sin decimales.
%f	Número con decimales.
%c	Carácter ASCII.
Aunque existen otros tipos, estos son los más importantes.	

Código fuente

```

<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 11 WebEstilo Basico </title>
  </head>
  <body>
    <?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>");
    ?>
  </body>
</html>

```

[Ejecutar codigo](#)

Vemos la ejecución

Dirección <http://leliadoura.com.es/dai/phpwebestilo/manual/basico/ejemplos/11.php>

Puede fácilmente intercalar **texto** con números **3**

0
1
2
3
4
5
6
7
8
9

3.4. Manejo de cadenas

Dado el uso del lenguaje PHP el tratamiento de cadenas es muy importante, existen bastantes funciones para el manejo de cadenas, a continuación explicaremos las más usadas.

- `strlen(cadena)`. Nos devuelve el número de caracteres de una cadena.
- `split(separador,cadena)`. Divide una cadena en varias usando un carácter separador.
- `sprintf(cadena de formato, var1, var2...)`. Formatea una cadena de texto al igual que `printf` pero el resultado es devuelto como una cadena.
- `substr(cadena, inicio, longitud)`. Devuelve una subcadena de otra, empezando por inicio y de longitud longitud.
- `chop(cadena)`. Elimina los saltos de línea y los espacios finales de una cadena.
- `strpos(cadena1, cadena2)`. Busca la cadena2 dentro de cadena1 indicándonos la posición en la que se encuentra.
- `str_replace(cadena1, cadena2, texto)`. Reemplaza la cadena1 por la cadena2 en el texto.

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 12 WebEstilo Basico </title>
  </head>
  <body>
    <?php
      echo strlen("12345"),"<br>";
      $palabras=split(" ","Esto es una prueba");
      for ($i=0;$palabras[$i];$i++)
        echo $palabras[$i],"<br>";
      $resultado=sprintf ("8X5 = %d <br>",8*8);
      echo $resultado,"<br>";
      echo substr("Devuelve una subcadena de otra",9,3),"<br><br>";

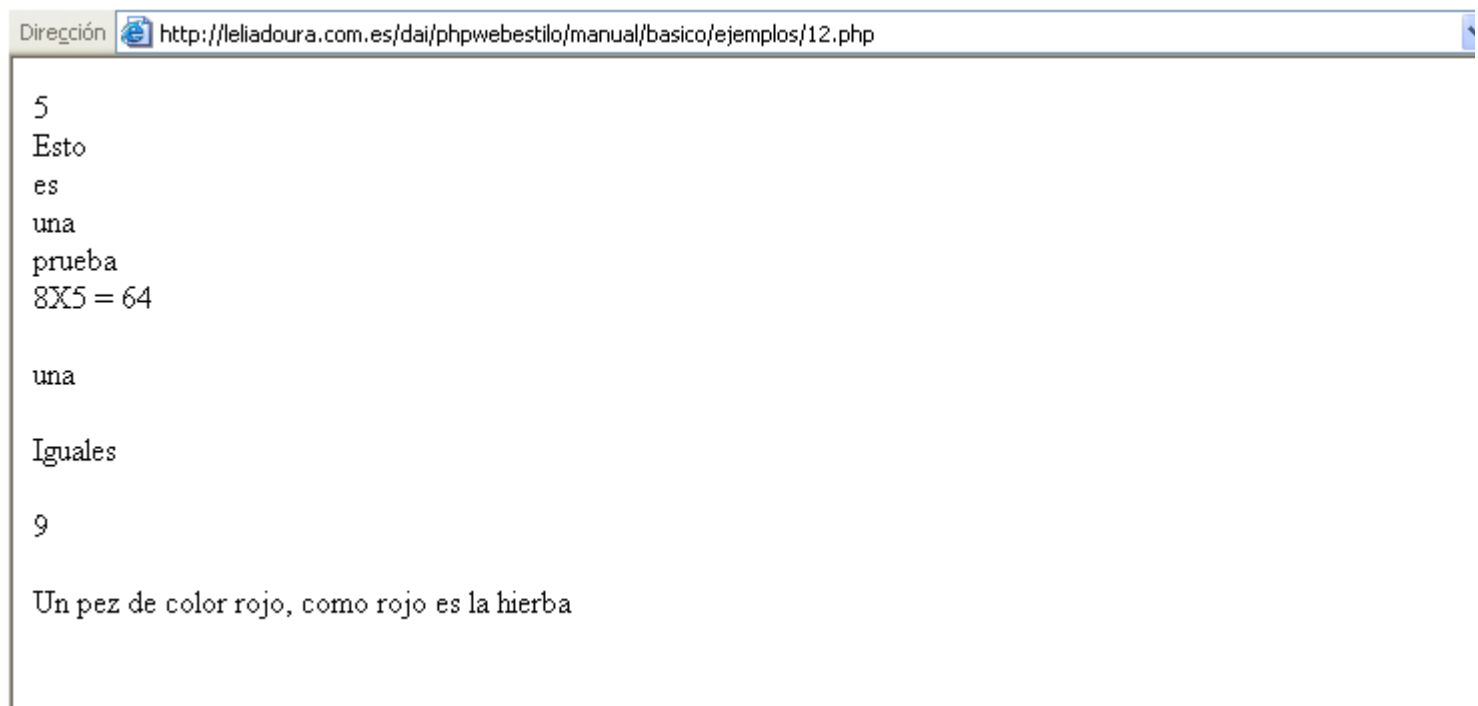
      if (chop("Cadena \n\n ")=="Cadena")
        echo "Iguales <Br><br>";

      echo strpos("busca la palabra dentro de la frase", "palabra"),"<br><br>";

      echo str_replace ("verde","rojo","Un pez de color verde, como verde es la
hierba"),"<br>";
    ?>
  </body>
</html>
```

[Ejecutar código](#)

Vemos la ejecución



```
Dirección http://leliadoura.com.es/dai/phpwebestilo/manual/basico/ejemplos/12.php
5
Esto
es
una
prueba
8X5 = 64

una

Iguales

9

Un pez de color rojo, como rojo es la hierba
```

4. Funciones

4.1. Funciones

El uso de funciones nos da la capacidad de agrupar varias instrucciones bajo un solo nombre y poder llamarlas a estas varias veces desde diferentes sitios, ahorrándonos la necesidad de escribirlas de nuevo.

```
<?php
function Nombre(parametro1, parametro2...)
{
    instrucción1;
    instrucción2;
    instrucción3;
    instrucción4;

    return valor_de_retorno;
}
?>
```

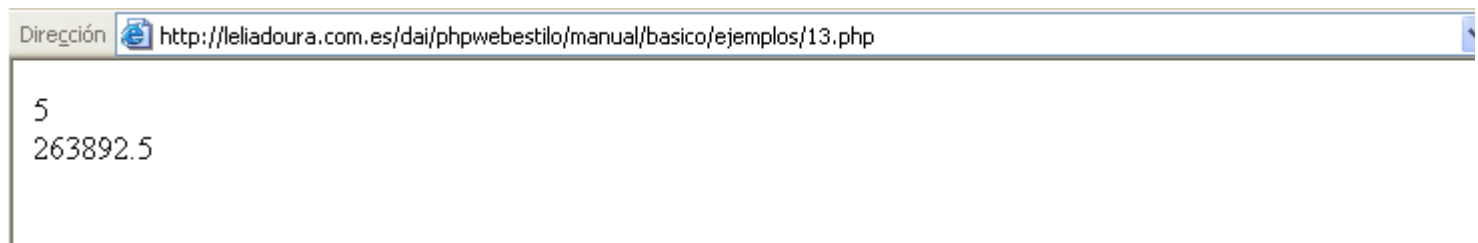
Opcionalmente podemos pasarle parámetros a las funciones que se trataran como variable locales y así mismo podemos devolver un resultado con la instrucción return valor; Esto produce la terminación de la función retornando un valor.

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
<head>
<title> Ejemplo 13 WebEstilo Basico </title>
</head>
<body>
<?php
function media_aritmetica($a,$b)
{
    $media=($a+$b)/2;
    return $media;
}
echo media_aritmetica(4,6),"<br>";
echo media_aritmetica(3242,524543),"<br>";
?>
</body>
</html>
```

[Ejecutar código](#)

Vemos la ejecución



4.2. Librerías

El uso de librerías es tremendamente útil, nos permiten agrupar varias funciones y variables en un mismo fichero, de manera que luego podemos incluir esta librería en distintas páginas y disponer de esas funciones fácilmente.

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<?php
// 14lib.php
```

```

function CabeceraPagina()
{
?>
    <FONT SIZE="+1">Esta cabecera estará en todas sus páginas.</FONT><br>
    <hr>
<?php
}
function PiePagina()
{
?>
    <hr>
    <FONT SIZE="-1">Este es el pie de página.</FONT><br>
    Autor: Joaquín Gracia
<?php
}
?>

```

Ahora vamos a crear 2 páginas que usan la librería definida anteriormente para conseguir que las dos paginas tengan la misma cabecera y pie de pagina.

La instrucción para incluir una librería en nuestra página es `include("nombre de librería")`

Código fuente

```

<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 15 WebEstilo Basico </title>
  </head>
  <body>
    <?php
      include("14lib.php");
    ?>
    <?php CabeceraPagina(); ?>

    Página del ejemplo: <FONT SIZE="+2">15.php</FONT>
    <br><br>


    Contenido blaba lkalk flkal <br><br>
    más cosas ..... <br><br>

    fin <br><br>
    <?php PiePagina(); ?>
  </body>
</html>

```

[Ejecutar código](#)

Vemos la ejecución

Dirección  <http://leliadoura.com.es/dai/phpwebestilo/manual/basico/ejemplos/15.php>

Esta cabecera estará en todas sus páginas.

Página del ejemplo: **15.php**

Contenido blaba lkalk flkal

más cosas

fin

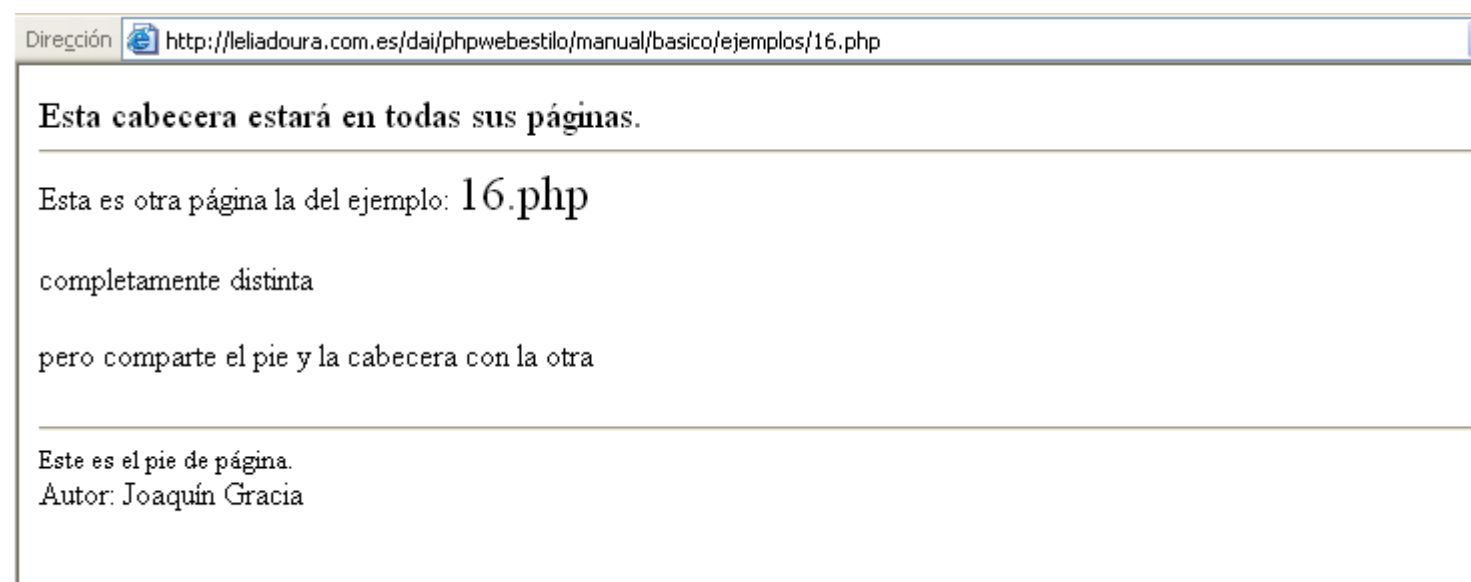
Este es el pie de página.
Autor: Joaquín Gracia

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 16 WebEstilo Basico </title>
  </head>
  <body>
    <?php
      include("14lib.php");
      CabeceraPagina();
    ?>
    Esta es otra página la del ejemplo: <FONT SIZE="+2">16.php</FONT>
    <br><br>
    completamente distinta<br><br>
    pero comparte el pie y la cabecera con la otra<br><br>
    <?php PiePagina(); ?>
  </body>
</html>
```

[Ejecutar código](#)

Vemos la ejecución



5. Ejemplos Prácticos

5.1. Formateo de página

En este ejemplo vamos a usar el PHP y la capacidad de definir librerías para conseguir que todas nuestras páginas tengan el mismo formato de página, incluyendo las partes comunes en librerías. Así mismo modificando la librería modificaríamos también todas las páginas de una manera muy rápida.

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<?php
    // 17lib.php
    function CabeceraPagina()
    {
    ?>
    <FONT SIZE="+1">Esta cabecera estará en todas sus páginas.</FONT><br>
    <hr>
```



```

<?php
}
function PiePagina()
{
?>
    <hr>
    <FONT SIZE="-1">Este es el pie de página.</FONT><br>
    Autor: Joaquín Gracia
<?php
}
function Indice()
{
?>
    <A HREF="18.php">Página 18 </A><BR>
    <A HREF="19.php">Página 19 </A><BR>
<?php
}
?>

```

Código fuente


```

<!-- Manual de PHP de WebEstilo.com -->
<html>
<head>
    <title> Ejemplo 18 WebEstilo Basico </title>
</head>
<body>
    <?php
        include("17lib.php");
    ?>
    <?php CabeceraPagina(); ?>
    <TABLE>
        <TR>
            <TD><?php Indice() ?></TD>
            <TD>
                Esta es otra página la del ejemplo: <FONT SIZE="+2">18.php</FONT>
                <br><br>
                completamente distinta<br><br>
                pero comparte el pie y la cabecera con la otra<br>
            </TD>
        </TR>
    </TABLE>
    <?php PiePagina(); ?>
</body>
</html>

```

[Ejecutar código](#)

Vemos la ejecución

Dirección  http://leliadoura.com.es/dai/phpwebestilo/manual/basico/ejemplos/18.php

Esta cabecera estará en todas sus páginas.

Esta es otra página la del ejemplo: **18.php**

[Página 18](#)
[Página 19](#) completamente distinta

pero comparte el pie y la cabecera con la otra

Este es el pie de página.
 Autor: Joaquín Gracia

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 19 WebEstilo Basico </title>
  </head>
  <body>
    <?php
      include("17lib.php");
    ?>
    <?php CabeceraPagina(); ?>
    <TABLE>
      <TR>
        <TD><?php Indice() ?></TD>
        <TD>
          Página del ejemplo: <FONT SIZE="+2">19.php</FONT>
          <br><br>

          Contenido blaba lkalk flkal <br>
          más cosas ..... <br><br>

          fin <br>
        </TD>
      </TR>
    </TABLE>
    <?php PiePagina(); ?>
  </body>
</html>
```

[Ejecutar código](#)

Vemos la ejecución



5.2. Enlace externo con frame

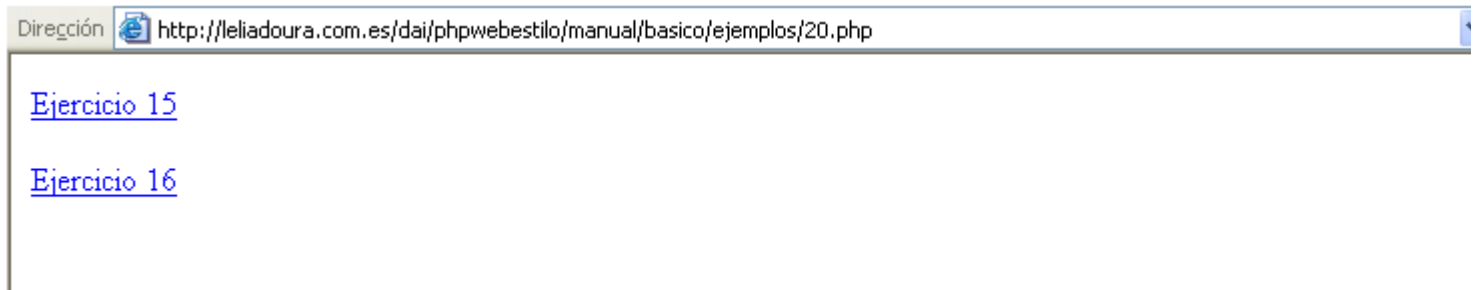
Con este ejemplo damos solución al problema de los enlaces externos y de forma que la web externa queda en la parte inferior del frame y así no se sale de nuestra web.

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 20 WebEstilo Basico </title>
  </head>
  <body>
    <A HREF="20frames.php?dire=15.php" > Ejercicio 15</A>
    <br><br>
    <A HREF="20frames.php?dire=16.php" > Ejercicio 16</A>
  </body>
</html>
```

[Ejecutar código](#)

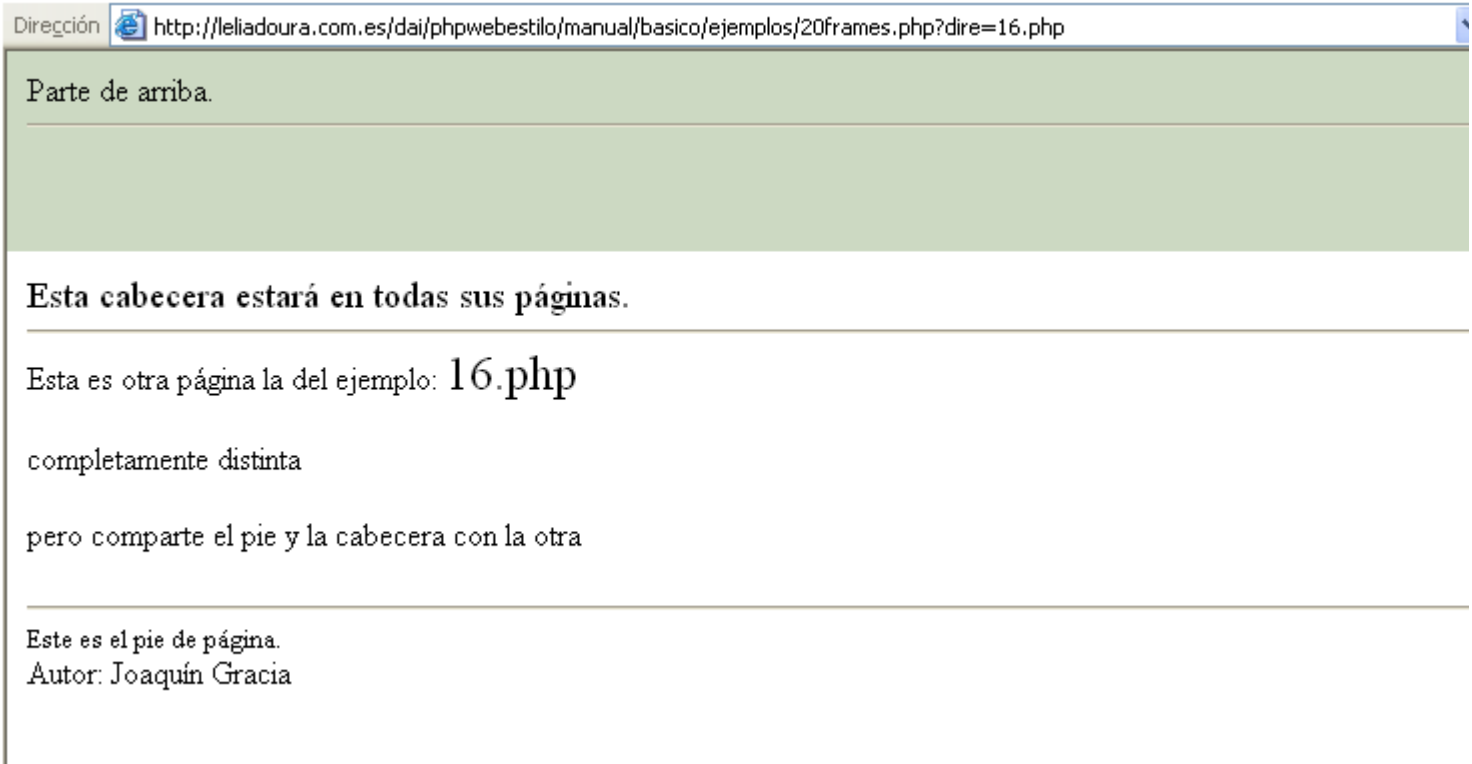
Vemos la ejecución



Si se pulsa en el enlace [Ejercicio 15](#) la ejecución es la siguiente:



Y si se pulsa en el enlace [Ejercicio 16](#) la ejecución es la siguiente:



Código fuente (fichero de las frames)

```
<!-- Manual de PHP de WebEstilo.com -->
<!-- frames -->
<html>
  <head>
    <title> Ejemplo 20frames WebEstilo Basico </title>
  </head>
  <frameset rows="100,*">
    <frame name="arriba" src="20arriba.php" marginwidth="10"
      marginheight="10" scrolling="auto" frameborder="0">
    <frame name="abajo" src="<?php echo $_GET[dire] ?>" marginwidth="10"
      marginheight="10" scrolling="auto" frameborder="0">
  </frameset>
</html>
```

Código fuente (parte de arriba de las frames)

```
<!-- Manual de PHP de WebEstilo.com -->
<html>
  <head>
    <title> Ejemplo 20arriba WebEstilo Basico </title>
  </head>
  <body BGCOLOR="#D1DDC8">

    Parte de arriba.<hr>

  </body>
</html>
```

Otro ejemplo en el que se da solución al problema de los enlaces externos.

Las novedades aquí son dos:

- En la frame de arriba se colocan los enlaces externos.
- En la frame de abajo, por defecto, se coloca la página de presentación (ver el código PHP de la frame de abajo)

Código fuente

```
<!-- Manual de PHP de WebEstilo.com -->

<html>
  <head>
    <title> Ejemplo 21 WebEstilo Basico </title>
  </head>
<!-- frames -->
<frameset rows="120,*">
  <frame name="arriba" src="21arriba.php" marginwidth="10"
    marginheight="10" scrolling="auto" frameborder="0">
  <frame name="abajo"
    src="<?php
      if ($_GET[dire]=="")
        echo "21presentacion.php"; // es la presentación
      else
        echo $_GET[dire];?>"
    marginwidth="10" marginheight="10" scrolling="auto" frameborder="0">
</frameset>
</html>
```

[Ejecutar código](#)

Vemos la ejecución

Dirección <http://leliadoura.com.es/dai/phpwebestilo/manual/basico/ejemplos/21.php>

[pagina de PRESENTACION](#)

[pagina 18](#) [pagina 19](#)

[Leliadoura](#) [Google](#)

Esta cabecera estará en todas sus páginas.

Página de PRESENTACION (21presentacion.php)

Contenido blaba lkalk flkal
 más cosas

fin

Este es el pie de página.
 Autor: Joaquín Gracia

Si se pulsa en el enlace [pagina 19](#) la ejecución es la siguiente:



Y si se pulsa en el enlace [Google](#) la ejecución es la siguiente:



Código fuente (parte de arriba de las frames)

```
<!-- Manual de PHP de WebEstilo.com -->
```

