

## Uniando cadenas

### La concatenación de cadenas

Para concatenar (unir), en una sola, varias *porciones* de texto hemos venido utilizando –en las instrucciones *print* y *echo*– un punto (.).

### El operador .

Este *punto* es un elemento muy importante que, además de la forma que hemos visto en las páginas anteriores, tiene los siguientes usos:

### Unir dos cadenas y recogerlas en una variable

Con la sintaxis:

```
$a="cad1" . "cad2";
```

o mediante

```
$a= $b . $c
```

podemos obtener una nueva variable formada por la *unión* dos trozos. Pero seguramente te preguntarás *¿qué ocurre si juntamos una variable cadena y una numérica? o ¿qué ocurre si juntamos dos variables numéricas?*

En cualquiera de los supuestos –puedes verlo en el ejemplo– las variables serán tratadas por PHP –con independencia de lo que puedan contener– como de tipo *cadena* y la variable que contiene el resultado es del tipo **string**.

### Añadir contenidos a una variable tipo *string*

Si utilizamos una sintaxis como esta:

```
$a .="cad1"  
o de este otro tipo  
$a .=$b
```

(presta mucha atención al punto que va delante del signo igual) se *añadiría* al **valor actual** de la variable **\$a** el contenido indicado después del signo igual.

Fíjate en la importancia del punto. Si está presente, se *añaden* nuevos contenidos a la variable; pero en el caso de que no lo estuviera, se *sustituirían* esos contenidos, con lo cual se perdería la información que esa variable pudiera contener.

Aquí tienes un ejemplo de concatenación de variables tipo *string*.

```
<?
#definamos y asignemos valores a variables tipo cadena
$cadena1="Esto es una cadena de texto";
$cadena2="Esta es una segunda cadena de texto";

#hagamos lo mismo con variables numéricas
$cadena3=127;
$cadena4=257.89;

# unámoslas mezclando tipos

$union1=$cadena1 . $cadena2;
$union2=$cadena1 . $cadena3;
$union3=$cadena3 . $cadena4;

#veamos que ha ocurrido

echo $union1,"<br>";
echo $union2,"<br>";
echo $union3,"<br>";

# modifiquemos ahora una cadena
# añadiéndole contenidos

$cadena3 .=" Este es el texto que se añadirá a la variable cadena3";
# imprimamos los resultados
echo $cadena3,"<br>";
# añadamos ahora un nuevo trozo, esta vez
# a partir de una cadena escrita con la
# sintaxis de documento incrustado
$cadena3 .=" <<<Pepito
  Ahora le añado a la cadena
  este trocillo asignado con el "formato"
  de documento incrustado
  Pepito;
# visualicemos el resultado
echo $cadena3,"<br>";
?>
```

[ejemplo18.php](#)

### ¡Cuidado!

Observa en el ejemplo que, excepcionalmente, la *sintaxis de documento incrustado* permite introducir comillas (sin ningún método especial), pero recuerda que en cualquier otro caso hay que recurrir al truco del que hablábamos en [aquí](#).

### Ejercicio nº 13

Utiliza los dos métodos de creación de variables de cadena y lo relativo al tratamiento de caracteres especiales para crear un script –al que llamaremos **ejercicio13.php**– que permita explicar el significado de las etiquetas <BODY>, <HEAD> y <HTML> Resalta, con comillas, las palabras de tu explicación que consideres más importantes.

