

Capítulo 4

Funciones y estructuras de control

1. Las condiciones

1.1 If

Las instrucciones del tipo condición permiten ejecutar el código si una condición es verdadera.

Por ejemplo, si el nombre es Roberto, muestre «Bienvenido»:

```
<?php
$nombre = 'Roberto'; //declaración de la variable $nombre

if ($nombre == 'Roberto') //comprueba la variable $nombre
{
    echo 'Bienvenido';
}
?>
```

El valor Roberto se asigna a la variable \$nombre y se comprueba con ayuda de la instrucción **if**.

Por tanto, la sintaxis es:

```
if (condition) { instrucción }
```

Tenga en cuenta que:

- «es igual a» se designa `==`
- «es diferente de» se designa `!=`
- «es inferior a» se designa `<`
- «es superior a» se designa `>`
- «es inferior o igual a» se designa `<=`
- «es superior o igual a» se designa `>=`

La instrucción **si no** se designa con **else**.

Por ejemplo, si el nombre es igual a Roberto, se muestra «Bienvenido», o «Hasta pronto» en cualquier otro caso:

```
<?php

$nombre = 'Pepe'; //declaración de la variable $nombre

if ($nombre == 'Roberto') //comprueba la variable $nombre
{
    echo 'Bienvenido';
}
else
{
    echo 'Hasta pronto';
}
?>
```

Da como resultado:

Hasta pronto

El código comprueba si la variable `$nombre` es Roberto, pero como no es el caso, el código entra en **else** (si no), y ejecuta `echo "Hasta pronto";`.

La última instrucción en las condiciones es **else if**, llamada **si no si**.

Esto permite probar otras condiciones que no han sido comprobadas por **if**.

Por ejemplo, si el nombre es Roberto, se muestra «Bienvenido»; si no, si es Pepe se muestra «Hola»; si no, «Hasta pronto»:

```
<?php

$nombre = 'Pepe'; //declaración de la variable $nombre

if ($nombre == 'Roberto') //comprueba la variable $nombre
{
    echo 'Bienvenido';
}
else if ($nombre == 'Pepe') //comprueba la variable $nombre
{
    echo 'Hola';
}
else
{
    echo 'Hasta pronto';
}

?>
```

Da como resultado:

Hola

El código comprueba si \$nombre es igual a Roberto y también si es igual a Pepe.

Como efectivamente es igual a Pepe, el código ejecuta `echo "Hola";`.

Puede añadir tantas instrucciones **else if** como quiera.

Si la condición (`$nombre == "Roberto"`) es cierta, las otras condiciones **else if** ni siquiera se comprueban, por tanto si entra en **if** se asegura de no pasar nunca por **else if** ni **else**.

Por el contrario, si escribe:

```
<?php

$nombre = 'Roberto'; //declaración de la variable $nombre

if ($nombre == 'Roberto') //comprueba la variable $nombre
{
```

```
        echo 'Bienvenido';
    }

    if ($nombre == 'Pepe') //comprueba la variable $nombre
    {
        echo 'Hola';
    }
    else
    {
        echo 'Hasta pronto';
    }
    ?>
```

El código comprueba si \$nombre es igual a Roberto y si también es igual a Pepe, pero como \$nombre se inicia con el valor Roberto, este código muestra «Bienvenido» y «Hasta pronto».

También puede anidar los **if** unos dentro de otros tantas veces como quiera.

Por ejemplo:

```
<?php

$edad = 30; //declaración de la variable $edad

if ($edad > 20) //comprueba la variable $edad
{
    if ($edad == 30) //comprueba la variable $edad
    {
        echo 'Bienvenido';
    }
    else {
        echo 'Hasta pronto';
    }
}
?>
```

Da como resultado:

Bienvenido

El programa comprueba si la variable `$edad` es superior a 20, y, como es el caso, el programa comprueba si `$edad` es igual a 30 y ejecuta `echo "Bienvenido";`.

Para terminar, puede intercalar el código HTML entre las condiciones en PHP.

Por ejemplo:

```
<?php

$nombre = 'Roberto'; //declaración de la variable $nombre

if ($nombre == 'Roberto') //comprueba la variable $nombre
{
    ?>
    Hola <!--Código HTML-->
    <?php
}
else
{
    ?>
    Hasta pronto <!--Código HTML-->
    <?php
}
?>
```

Es igual a:

```
<?php

$nombre = 'Roberto'; //declaración de la variable $nombre

if ($nombre == 'Roberto') //comprueba la variable $nombre
{
    echo 'Hola';
}
else
{
    echo 'Hasta pronto';
}
?>
```

1.2 switch

La instrucción **switch** es igual a **if**, pero el programador la utiliza para comprender mejor el código. Si se utiliza un **break** para salir del **switch**, entonces es igual a **if**. De lo contrario, se ejecutarán todas las instrucciones del **case** en el que ha entrado.

La sintaxis es:

```
switch (condición) {  
    case expresión: instrucción  
    case expresión: instrucción  
    ...  
}
```

```
<?php
```

```
$nombre = 'Roberto'; //declaración de la variable $nombre
```

```
switch ($nombre) //comprueba la variable $nombre
```

```
{
```

```
    case 'Roberto':
```

```
        echo 'Hola';
```

```
        break;
```

```
    case 'Juan':
```

```
        echo 'Hasta pronto';
```

```
        break;
```

```
}
```

```
?>
```

La instrucción **break** provoca la salida del **switch** y si **\$nombre** es igual a "Roberto" el código ejecutará **echo "Hola"** y **break**, y saldrá del **switch** sin comprobar "Juan".

Eso es igual a la instrucción **else if** en lugar de **if**:

```
<?php
```

```
$nombre = 'Roberto'; //declaración de la variable $nombre
```

```
if ($nombre == 'Roberto') //comprueba la variable $nombre
```

```
{
```

```
    echo 'Hola';
```

```
}
```