

# Variables, Operadores y Arrays



# Variables

- Las representamos por un nombre precedido por el símbolo \$
  - `$nombre = "Pepe";`
  - `$apellidos = "Sánchez Martínez";`
  - `$peliculas = array("Superman", "Superman 2");`
- Son sensibles a mayúsculas y minúsculas
  - `$Nombre`
  - `$nombre`
  - `$nOmbre`



# Ejemplos de Variables

```
$nombre = "Pepe";           //Asignamos una cadena

$edad = 26;                  //Asignamos un número entero

$texto = "Mi nombre es $nombre"; //Cadena con
variable

$casado = FALSE;             //Valor booleano

$_var = 10.22                 //Asignamos un número real

$4var = 'Juan';               //ERROR, comienzan por letra
o _

$concatena = $nombre . ' ' . $edad;
```



## VARIABLES DEL SISTEMA

- `$_POST`
- `$_GET`
- `$_SERVER`
- `$_SESSION`
- ...

## BUENAS PRÁCTICAS

- Nombres descriptivos
- Debemos tener en cuenta qué contiene cada variable



# OPERADORES

## OPERADORES DE ASIGNACIÓN

= El operador de igual ya lo hemos visto usar y es fácil de entender. La parte de la derecha es asignada a la variable que aparece en la izquierda.

## OPERADORES DE CONCATENACIÓN

. El operador usado para la concatenación de cadenas es el carácter punto "."



# OPERADORES

## OPERADORES DE REFERENCIA

**&** Este operador es el de referencia. Con él le decimos a una variable que tome el valor de otra variable.

### Ejemplo

```
$a = 5;  
$b = &$a;  
echo $b;  
$a = 9;  
echo $b;
```



# OPERADORES

## OPERADORES ARITMÉTICOS

+	Suma
-	Resta
*	Producto
/	División
%	Resto de la operación de división



# OPERADORES

## OPERADORES DE COMPARACIÓN

==	Igual
===	Igual y del mismo tipo
!=	Distinto
<	Menor que
>	Mayor que
<=	Menor o igual
>=	Mayor o igual

Estos operadores los utilizaremos mucho en las operaciones de control y bucles, que veremos después.



# OPERADORES

## OPERADOR TERNARIO

El operador ternario es '?' y sirve para indicar de forma abreviada un condicional.

Su sintaxis es:

condición ? valor\_verdadero : valor\_falso

Siendo:

- Condición, la condición que se evalúa.
- valor\_verdadero, lo que devuelve en caso de ser verdadera la condición.
- valor\_falso, lo que devuelve en caso de ser falsa la condición.

Ejemplo: `echo empty($x) ? "está vacía" : "No está vacía";`



# Arrays o vectores

Es una variable con un nivel de complejidad alto. Se puede decir que es una variable compuesta, ya que se compone de varios elementos. Podemos afirmar que es una forma de agrupar datos. Imaginemos la cola de un cine:

```
$cola = array("Juan", "Joaquín", "Javier",  
"Jesús");
```



# Arrays o vectores

Cuando hablamos de un array estamos agrupando por parejas de clave y valor.

Esto quiere decir que cada posición estará definida por un índice, y junto a este índice un valor.

```
$variablearray[clave] = valor;
```

Tipos de Arrays:

- Por índice
  - \$nombre[0] = "José";
  - \$nombre[1] = "Juan";
- Secuenciales
  - \$nombre[] = "José";
  - \$nombre[] = "Juan";
- Asociativos
  - \$capital["España"] = "Madrid";
  - \$capital["Francia"] = "París";



# Arrays o vectores

Formas de asignar valores a un array:

```
$nombres = array("Javier", "Jesús", "José", "Juan");  
$nombres = ["Javier", "Jesús", "José", "Juan"];
```

La segunda de las dos formas anteriores es más óptima en tiempo de ejecución.

Otra forma es indicar las parejas de clave y valor:

```
$nombres = array( 1 => "Javier", 2 => "Jesús");  
$capitales = [ "españa" => "madrid", "francia" => "parís");
```

Si queremos un valor concreto del array:

```
$nombres[2];  
$capitales["francia"];
```