

## Tipos de datos

El tipo de datos que se va a utilizar en una variable normalmente no lo indica el programador; en su lugar, lo decide PHP en tiempo de ejecución dependiendo del contexto en el que se utilice esa variable, es decir; dependiendo del valor o resultado de la expresión que se le asigna.

### Booleanos

Este es el tipo más simple. Un boolean expresa un valor lógico. Puede ser true (verdadero) o false (falso).

### Sintaxis

Para especificar un literal boolean, se usan alguna de las palabras clave TRUE o FALSE. Ambas son insensibles a mayúsculas y minúsculas.

```
<?php  
$Correcto = true; // Asigna true a la variable $Correcto  
?>
```

Un valor es considerado como FALSE cuando:

- el boolean **FALSE** mismo
- el integer 0 (cero)
- el float 0.0 (cero)
- el valor string vacío, y el string "0"
- un array con cero elementos
- el tipo especial **NULL** (incluidas variables no establecidas)

Cualquier otro valor será considerado como TRUE.

Ejercicio:

1. Crear un archivo PHP de nombre tipoDatoBooleano.php
2. Escribir el siguiente código:

```
<?php  
$miVariableTipoBooleano = 0;  
if ($miVariableTipoBooleano){  
    echo "Verdadero";  
}else{  
    echo "Falso";  
}  
?>
```

3. Visualizar el contenido en tu navegador favorito
4. Revisar el código de la página dinámica.

### Enteros (integer)

Los enteros se pueden especificar usando una de las siguientes sintaxis:

```
<?php  
$a = 1234; // número decimal  
$a = -123; // un número negativo  
$a = 0123; // número octal (equivalente a 83 decimal)  
$a = 0x1A; // número hexadecimal (equivalente a 26 decimal)  
?>
```

**Nota:** Debido a que PHP asigna los tipos en función del contexto en el que se utilizan, la combinación de un tipo entero con un tipo string puede dar como resultado otro entero, si el string se evalúa en un contexto numérico.

### Ejercicio:

1. Crear un archivo PHP de nombre tipoDatosEntero.php
2. Escribir el siguiente código:

```
<?php  
$edad = 40;  
echo "Mi edad actual es: $edad";  
?>
```

3. Visualizar el contenido en tu navegador favorito
4. Revisar el código de la página dinámica.
5. **Actualización de código:** Agregar una variable que se llame \$anio que sea igual a 2020, mostrar en la línea de abajo el texto: Mi edad para el año 2021 será 41. TIP: Utilizar el operador aritmético de suma.

### Nº de punto flotante (float)

Los números de punto flotante (también conocidos como "flotantes", "dobles" o "números reales") pueden ser especificados usando cualquiera de las siguientes sintaxis:

```
<?php  
$a = 1.234;  
$b = -1.23;  
?>
```

Ejercicio:

1. Crear un archivo PHP de nombre tipoDatosFlotante.php
2. Escribir el siguiente código:

```
<?php  
$estatura = 1.71;  
$peso = 67.5;  
echo "Mi estatura actual es: $estatura ";  
echo "<br>Mi peso actual es: $peso";  
?>
```

3. Visualizar el contenido en tu navegador favorito
4. Revisar el código de la página dinámica.
5. **Actualización de código: Utiliza la fórmula del índice de masa corporal**
  - a.  $IMC = \text{Peso (kg)} / \text{altura (m)}^2$
6. Imprime la siguiente línea: Mi IMC es: [resultado]

**TIP: Utilizar el operador aritmético de multiplicación, división y los paréntesis para prioridades.**

### Cadenas de caracteres (string)

Un string es una serie de caracteres donde un carácter es lo mismo que un byte. Esto significa que PHP solo soporta el conjunto de 256 caracteres.

Un string puede llegar a almacenar hasta 2Gb de tamaño.

Un valor de tipo string (literal) puede ser especificado de dos formas diferentes:

- *comillas simples '...'*
- *comillas dobles "..."*

#### *Comillas simples*

Es la forma más sencilla de representar un string y la que consume menos ciclos de CPU para su representación, por lo que es la recomendada cuando sólo vamos a manipular cadenas de caracteres puros sin variables ni códigos de escape.

Para especificar una comilla simple dentro de un string encerrado entre comillas simples, tendremos que escribirlo anteponiendo el carácter de barra invertida (\). Para especificar una barra invertida literal, se duplica (\\\).

#### Ejemplo

```
<?php  
// Cadena de caracteres  
$cadena='Cadena de caracteres';  
// Me llamo 'Luis Roca' y mi web es minubeinformatica.com  
$cadena='Me llamo \'Luis Roca\' y mi web es minubeinformatica.com';  
// El archivo se encuentra en c:\wamp64\www  
$cadena='El archivo se encuentra en c:\\wamp64\\www';  
?>
```

Ejercicio: Escribir el ejemplo anterior en un archivo PHP de nombre comillasSimples.php y visualizarlo en el navegador de tu preferencia.

Modificaciones: mostrar los resultados de cada valor de la variable \$cadena.

...