

Sintaxis PHP

Es un lenguaje de programación que se utiliza para el desarrollo web, por sus siglas Hipertext Preprocesor. Es un lenguaje principalmente utilizado para trabajar con HTML, para crear contenido dinámico y bases de datos. Actualmente PHP como es un lenguaje del lado del servidor se ha quedado en backend pero es muy popular en los CMS.

Entre sus características más resaltantes:

- Es independiente de plataforma.
- Es un lenguaje del lado del servidor (server-side).
- Es un lenguaje interpretado.
- Ofrece alto rendimiento.
- El lenguaje es de concepción rápida.
- Contiene gran soporte, está muy documentado y posee una gran librería de funciones.
- Soporta la programación orientada a objeto y estructurada.
- Puede expandirse a módulos externos.

Sintaxis

Para incrustar el código en los documentos html debemos realizar los siguiente:

- Imprimir

```
<?php  
echo"Hola Mundo Uniempresarial"  
?>
```

Etiquetas recomendadas:

- ```
<?php ?>
```

 // Etiqueta de apertura y cierre de PHP (la más segura y compatible).
- Evitar 

```
<? ?>
```

 (short tags), porque pueden estar deshabilitadas.

## Tipos primitivos o escalares de datos en PHP

| Tipo               | Valor                            | Ejemplo                                                                 |
|--------------------|----------------------------------|-------------------------------------------------------------------------|
| boolean            | True - false                     | <code>\$bandera = true;</code><br><code>echo gettype(\$bandera);</code> |
| Entero             | Valores enteros                  | <code>\$numero = 8;</code><br><code>\$numero = 0;</code>                |
| Decimal / flotante | Valores con soporte de decimales | <code>\$nota = 4.5;</code><br><code>\$promedio = 3.5;</code>            |

|                  |                                   |                           |
|------------------|-----------------------------------|---------------------------|
| Cadenas          | Cadenas de caracteres             | \$nombre = "Juan Andres"; |
| Nulo             | null                              | \$miVar = null;           |
| Array o arreglos | Colecciones de datos o listas     | \$miArray = [];           |
| Objeto           | A partir de la creación de clases | class Persona {}          |

| Operadores Aritméticos |                             |                        |
|------------------------|-----------------------------|------------------------|
| +                      | Suma                        | \$x + \$y              |
| -                      | Resta                       | \$x - \$y              |
| *                      | Multiplicación              | \$x * \$y              |
| /                      | División                    | \$x / \$y              |
| %                      | Módulo                      | \$x % \$y              |
| **                     | Potencia                    | \$x ** \$y             |
| ++                     | Incremento                  | \$x++                  |
| --                     | Decremento                  | \$x--                  |
| +=                     | Suma y asignación           | \$x += \$y             |
| -=                     | Resta y asignación          | \$x -= \$y             |
| *=                     | Multiplicación y asignación | \$x *= \$y             |
| /=                     | División y adignación       | \$x /= \$y             |
| =                      | Asignación                  | \$x = 10;              |
| .                      | Concatenar                  | echo 'hola: '.\$nombre |

| Operadores Relacionales      |       |                       |             |       |
|------------------------------|-------|-----------------------|-------------|-------|
| ==                           |       | Comparación           | \$x == \$y  |       |
| >                            |       | Mayor                 | \$x > \$y   |       |
| <                            |       | Menor                 | \$x < \$y   |       |
| >=                           |       | Mayor igual           | \$x >= \$y  |       |
| <=                           |       | Menor igual           | \$x <= \$y  |       |
| !=                           |       | Diferente             | \$x != \$y  |       |
| <>                           |       | Diferente             | \$x <> \$y  |       |
| ===                          |       | Identico              | \$x === \$y |       |
| Operadores Lógicos booleanos |       |                       |             |       |
| &&                           |       | AND o Y               |             |       |
|                              |       | OR o O                |             |       |
| ?:                           |       | If-then-else ternario |             |       |
| \$X                          | \$Y   | \$X    \$Y            | \$X && \$Y  | !\$X  |
| True                         | True  | True                  | True        | False |
| True                         | False | True                  | False       | False |
| False                        | True  | True                  | False       | True  |
| False                        | False | False                 | False       | True  |

## Verificación de tipos de variables

Declaración de Variables: al igual que otros lenguajes de programación PHP también tiene varios tipos de variables como:

- Variables: Enteras, reales y de texto (cadenas de caracteres).

```
<?php
$entera= 50;
$decimales= 13.56;
$texto1="Variable de texto";
?>
```

- Constantes: son valores que no cambian con la ejecución del programa.

```
<?php
define('IVA', '19.01');
echo(IVA);
?>
```

## Declaración con definición estricta

PHP 7, puedes declarar el tipo de datos que deben recibir los argumentos y devolver las funciones. Esto mejora la seguridad del código, facilita su lectura y previene errores. En los escalares disponibles se tienen los enteros(int), decimales(float), cadenas de texto o caracteres(string) y booleanos(bool), dentro de los compuestos array, object, callable, iterable, void y mixed

```
<?php
declare(strict_types=1);
function sumar(int $number1, int $number2): int {
 return $number1 + $number2;
}
// Esto genera TypeError por tipado estricto
echo sumar(2.5, 3.5);
?>
```

| Situación                     | Resultado                      |
|-------------------------------|--------------------------------|
| Declarar variable y no usarla | OK, sin error                  |
| Usar variable no declarada    | NOTICE (Undefined variable)    |
| Error en tipo de argumento    | TypeError (con strict_types=1) |

| Sentencias de Control o Selección |                                                                                                                                                                                                |                                                                                                                                                                                                                                                                                                                                                    |
|-----------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Sentencia                         | Sintaxis                                                                                                                                                                                       | Ejemplo                                                                                                                                                                                                                                                                                                                                            |
| if                                | <pre>if (condición){     sentencia; }</pre>                                                                                                                                                    | <pre>\$num1 = 3; if (\$num1 &lt; 10){     print ("num1 es menor a"); }</pre>                                                                                                                                                                                                                                                                       |
|                                   | <pre>if (condición){     Instrucción 1;     .....     Instrucción n; } else {     Instrucción 1;     .....     Instrucción n; }</pre>                                                          | <pre>\$num1 = 3.5; \$num2 = 4.5; if (\$num2 &gt; \$num1){     \$resta = \$num2 - \$num1;     echo ("La resta es:". \$resta); }else{     \$suma = \$num1 + \$num2;     echo ("La suma es:". \$suma); }</pre>                                                                                                                                        |
|                                   | <pre>if (condición){     Instrucción 1;     .....     Instrucción n; } elseif {     Instrucción 1;     .....     Instrucción n; }else{     Instrucción 1;     .....     Instrucción n; }</pre> | <pre>&lt;?php     \$edad = 5;     if (\$edad &gt; 17): ?&gt;         &lt;p&gt;mayor de edad&lt;/p&gt;     &lt;?php elseif(\$edad &gt; 65): ?&gt;         &lt;p&gt;Adulto mayor&lt;/p&gt;     &lt;?php else: ?&gt;         &lt;h1&gt;Es menor de edad&lt;/h1&gt;     &lt;?php endif ?&gt;</pre>                                                     |
| ?:<br>(ternario)                  |                                                                                                                                                                                                | <pre>\$resultado = (\$var1 &gt; \$var3) ? "Var1 es mayor" : "Var1 es menor"; echo \$resultado."&lt;br/&gt;";</pre>                                                                                                                                                                                                                                 |
| ??<br>(Null<br>Coalescing)        |                                                                                                                                                                                                | <pre>&lt;?php declare(strict_types=1); // Usando operador Null Coalescing con tipado estricto function obtenerNombre(?string \$nombre): string {     return \$nombre ?? 'Invitado'; } // Simular \$_GET con tipado estricto \$nombreIngresado = \$_GET['nombre'] ?? null; echo obtenerNombre(\$nombreIngresado); // resultado Invitado ?&gt;</pre> |
| switch                            | <pre>switch(condición){     case valor1:         instrucciones...         break;</pre>                                                                                                         | <pre>switch(\$i){     case 1:         instrucciones...         break;</pre>                                                                                                                                                                                                                                                                        |

|                       |                                                                                                                                                                                                                                    |                                                                                                                                                                                     |
|-----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
|                       | <pre> case valor2: instrucciones... break; default: instrucción por defecto; } </pre>                                                                                                                                              | <pre> case 2: instrucciones... break; default: instrucción por defecto; } </pre>                                                                                                    |
| while                 | <pre> [inicialización] while (condición de terminación) {     cuerpo;     iteración; } </pre>                                                                                                                                      | <pre> \$i=1; while (\$i &lt;= 10) {     cuerpo;     i++; } </pre>                                                                                                                   |
| Variación while       | <pre> \$i = 1; while (\$i &lt;= 10):     echo \$i;     \$i++; endwhile; </pre>                                                                                                                                                     |                                                                                                                                                                                     |
|                       | <pre> [inicialización] do {     cuerpo;     iteración; } while (condición de terminación); </pre>                                                                                                                                  | <pre> \$i=1; do{     if (\$i % 2 == 0)         echo("numero par");     \$i++; } while (\$i&lt;=10); </pre>                                                                          |
| for                   | <pre> for (inicialización; condición de fin; iteracion){     cuerpo; } </pre>                                                                                                                                                      | <pre> for (\$i = 1; \$i &lt;= 10; \$i++){     if (\$i % 2 ==0)         print("numero par".\$i); } </pre>                                                                            |
| Variación del for     | <pre> for (\$i = 1; ; \$i++){     if (\$i % 2 ==0)         print("For: numero par".\$i."&lt;br/&gt;");     if (\$i &gt; 10) {         # code...         break;     } } </pre>                                                      | <pre> for (\$i = 1; ; ){     if (\$i % 2 ==0)         print("For: numero par".\$i."&lt;br/&gt;");     if (\$i &gt; 10) {         # code...         break;     }     \$i++; } </pre> |
| Foreach (solo arrays) | <p>Sintaxis: foreach (\$array: \$valor )</p> <pre> \$myarray = array(2, 4, 6, 8, 10, 12); foreach (\$myarray as &amp;\$valor) {     echo \$valor; } </pre>                                                                         |                                                                                                                                                                                     |
| break                 | Finaliza la ejecución de una estructura de control repetitiva como: while, for, foreach, do-while y switch.                                                                                                                        |                                                                                                                                                                                     |
| continue              | Al igual del break se utiliza en estructura repetitivas e indica continuar saltar el resto de la iteración actual del bucle y continuar la ejecución en la evaluación de la condición, para luego comenzar la siguiente iteración. |                                                                                                                                                                                     |

```
$myarray = array(2, 4, 6, 8, 10, 12);
foreach ($myarray as &$valor) {
 echo ($valor. "
");
 continue;
 echo("Despues del continue");
 $valor * 2;
}
echo "Fuera del for";
```