

Módulo 4. PHP



**CODE
SPACE**
ACADEMY

Carlos Herrera

Índice

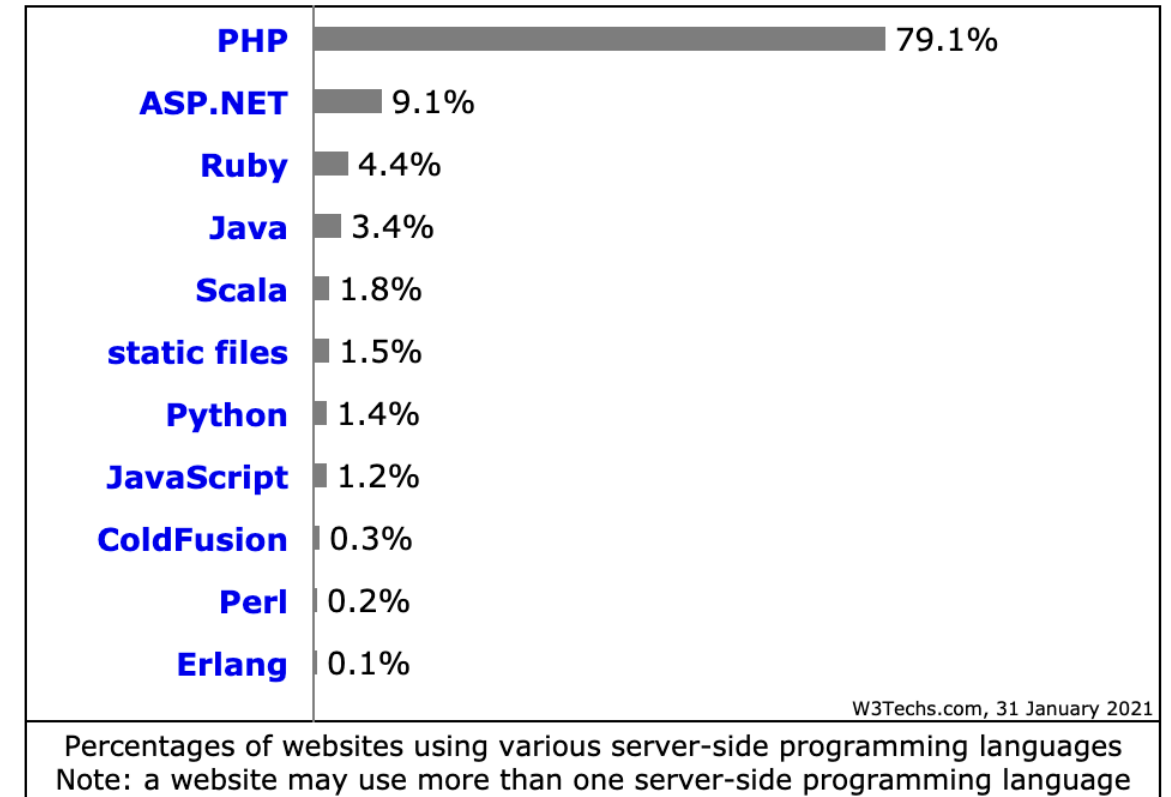
- ¿Qué es PHP?
- Aspectos previos
- Variables y constantes. Tipos de datos.
- Operadores
- Sentencias de control (if, for, switch...)
- Funciones

1. ¿Qué es PHP?

PHP es un lenguaje de programación de uso general que se adapta especialmente al desarrollo web. Data de 1995.

El código PHP suele ser procesado en un servidor web para generar contenido web (normalmente html o json).

Fuente: Wikipedia.



Fuente: https://w3techs.com/technologies/overview/programming_language.

2. Aspectos previos

1. Un archivo php contiene líneas para procesar y líneas para no procesar. Las líneas para procesar se delimitan con `<?php` y `?>`.
2. Los punto y coma o semicolon(;) son obligatorios.
3. Si se produce un error en la programación y la excepción no es capturada, la ejecución se interrumpe inmediatamente.
4. Los comentarios de una sola línea con `//` y de múltiples líneas con `/* Comentario */`
5. Primera posición de los string y de los array es 0.
6. Los strings se definen entre `'` o `"`.
7. Podemos hacer una depuración básica a través de `echo`, `print_f()`, `var_dump()`, etc.

3. Variables y constantes

Reglas para crear variables y constantes:

- Las variables deben empezar con \$ seguido del nombre de la variables. Las constantes no empiezan por \$.
- El nombre de la variable debe empezar con una letra o un guión bajo. No puede empezar por un número.
- En los nombres de las variables solamente se permiten caracteres alfanuméricos y el guión bajo (A-z,0-9,_)
- Distingue entre mayúsculas y minúsculas.
- No pueden ser una palabra reservada.
- *Los nombres deben tener significado.*
- *Convención de camelCase para variables y SNAKE_CASE para constantes.*

3. Variables y constantes

```
<?php

// Variables primitivas
$string = "cadena de texto";
$integer = 27;
$float = 22.83;
$bool = false;

// Constantes
define("OBJETO", "Soy una tetera");
const OTRO_OBJETO = "Soy un cucharón";
```

3. Variables y constantes: Arrays

```
// Arrays
$array = array('dato', 5, 21.2, true); // Sintaxis antigua
$array2 = ['dato', 5, 21.2, true];     // Sintaxis PHP 5.4
$array_asociativo = [
    'nombre' => 'Carlos',
    'apellidos' => 'Herrera Conejero',
    'telefono' => '600123456',
    'email' => 'carlos@correo.com',
    'ciudad' => 'Málaga'
];
```

3. Variables y constantes: Clases

```
$player1 = new Archer('Simtra');  
  
assertEquals('Simtra', $player1->getName());
```

```
<?php  
  
class Archer {  
    private string $name;  
    private int $attack;  
    private int $shield;  
    private int $lifePoints;  
  
    public function __construct(string $name) {  
        $this->name = $name;  
        $this->attack = 5;  
        $this->shield = 1;  
        $this->lifePoints = 13;  
    }  
  
    public function getName(): string {  
        return $this->name;  
    }  
  
    public function attack($enemy): void {  
        // ...  
    }  
  
    public function healSelf($healPoints) {  
        $this->lifePoints += $healPoints;  
    }  
}
```


4. Operadores

```
<?php
// Operadores aritméticos:
$suma = 5 + 8;
$resta = 3 - 2;
$multiplicar = 2 * 5;
$dividir = 4 / 2;
$resto = 20 % 5;
```

```
// Concatenación
$nombreCompleto = 'Carlos ' . 'Herrera Conejero';
```

```
// valores para incremento y decremento:
$numero++;
++$numero;
--$numero;
$numero--;

// Asignar operación:
$numero += 5;
$numero -= 10;
$numero *= 20;
$numero /= 5;
```

4. Operadores

```
// Operadores lógicos
assertEquals(true, 15 > 7, 'Mayor que');
assertEquals(false, 15 < 7, 'Menor que');
assertEquals(true, 15 >= 7, 'Mayor o igual que');
assertEquals(false, 15 <= 7, 'Menor o igual que');
assertEquals(true, 15 == 15, 'Igual que');
assertEquals(true, 15 == '15', 'Igual que');
assertEquals(true, 15 === 15, 'Estrictamente igual que');
assertEquals(false, 15 === '15', 'Estrictamente igual que');
```

```
// And, Or, Not
assertEquals(false, true && false, 'And');
assertEquals(true, true || false, 'Or');
assertEquals(false, !true, 'Not');
```

5. Sentencias de control. If

```
if ($coder) {  
    $resultado = "Eres Coder!";  
}  
  
if ($coder) {  
    $resultado = "Eres Coder!";  
} else {  
    $resultado = "Aprende a codear!";  
}
```

```
if ($curso == "fullstack") {  
    $resultado = "Eres del curso de desarrollo fullstack!";  
} else if ($codespace == "java") {  
    $resultado = "Eres del curso de Java";  
} else {  
    $resultado = "No estas registrado aún";  
}
```

5. Sentencias de control. If

```
// Operador ternario  
$frase = $edad >= 18 ? "Eres mayor de edad" : "Eres menor de edad";
```

```
// If alternativo  
if ($curso == "fullstack"):  
    $resultado = "Eres del curso de desarrollo fullstack!";  
elseif ($codespace == "java"):  
    $resultado = "Eres del curso de Java";  
else:  
    $resultado = "No estas registrado aún";  
endif;
```

5. Sentencias de control. Switch

```
<?php

switch($optionId){
    case (1):
        $optionText = "Leer menú";
        break;
    case (2):
        $optionText = "Editar menú";
        break;
    case (3):
        $optionText = "Eliminar menú";
        break;
    default:
        $optionText = "Opción no disponible";
        break;
}
```

5. Sentencias de control. Bucles

- **while** - repite un bloque de código mientras la condición sea cierta
- **do...while** - ejecuta un bloque de código sí o sí, y luego lo repite mientras la condición sea cierta
- **for** - repite un bloque de código mientras la condición sea cierta
- **foreach** - repite un bloque de código tantas veces como elementos tenga un array

Módulo 4 – Tema 1

PHP: INTRODUCCION



**CODE
SPACE**
ACADEMY

DESCANSO



**CODE
SPACE**
ACADEMY