

Capítulo 2: Conceptos Básicos de PHP

2.1 Variables y Tipos de Datos

Las variables en PHP se usan para almacenar datos y comienzan con un signo \$ seguido del nombre de la variable. PHP admite varios tipos de datos:

- **Cadena (String):** Una secuencia de caracteres, por ejemplo, "¡Hola, Mundo!"
- **Entero (Integer):** Números enteros, por ejemplo, 123
- **Flotante (Float):** Números con decimales, por ejemplo, 3.14
- **Booleano (Boolean):** Representa TRUE o FALSE
- **Arreglo (Array):** Una colección de valores, por ejemplo, [1, 2, 3, 4, 5]
- **Objeto (Object):** Instancias de clases
- **NULL:** Un tipo especial que representa una variable sin valor

Ejemplo de uso de variables:

php

CopyEdit

```
<?php

$cadena = "¡Hola, Mundo!";

$entero = 123;

$flotante = 3.14;

$booleano = true;

$arreglo = array(1, 2, 3, 4, 5);


echo $cadena; // Imprime: ¡Hola, Mundo!

echo $entero; // Imprime: 123

echo $flotante; // Imprime: 3.14

echo $booleano; // Imprime: 1 (true se representa como 1)

print_r($arreglo); // Imprime: Array ( [0] => 1 [1] => 2 [2] => 3 [3] => 4 [4] => 5 )

?>
```

2.2 Operadores y Expresiones

Los operadores se utilizan para realizar operaciones sobre variables y valores. Algunos operadores comunes son:

- **Aritméticos:** +, -, *, /, %

- **De Asignación:** =, +=, -=, *=, /=
- **De Comparación:** ==, !=, >, <, >=, <=
- **Lógicos:** && (AND), || (OR), ! (NOT)

Ejemplo:

php

CopyEdit

```
<?php
```

```
$a = 10;
```

```
$b = 5;
```

```
// Aritméticos
```

```
echo $a + $b; // Imprime: 15
```

```
// Comparación
```

```
if ($a > $b) {
```

```
    echo "a es mayor que b";
```

```
}
```

```
// Lógicos
```

```
if ($a > 5 && $b < 10) {
```

```
    echo "Ambas condiciones son verdaderas";
```

```
}
```

```
?>
```

2.3 Sentencias Condicionales

Las estructuras condicionales permiten ejecutar diferentes acciones según condiciones específicas. Las más comunes son **if**, **else** y **elseif**.

Ejemplo:

php

CopyEdit

```
<?php
```

```
$num = 10;
```

```
if ($num > 0) {  
    echo "El número es positivo."  
} elseif ($num < 0) {  
    echo "El número es negativo."  
} else {  
    echo "El número es cero."  
}  
?>
```

2.4 Bucles en PHP

Los bucles permiten ejecutar un bloque de código repetidamente mientras se cumpla una condición. Los bucles más comunes son: **for**, **while**, **do...while** y **foreach**.

Ejemplo de un bucle **for**:

php

CopyEdit

```
<?php  
for ($i = 0; $i < 5; $i++) {  
    echo "Valor de i: $i<br>";  
}  
?>
```

2.5 Funciones

Las funciones son bloques de código reutilizables que se pueden ejecutar en cualquier momento. Se definen con la palabra clave **function**.

Ejemplo:

php

CopyEdit

```
<?php  
function saludar($nombre) {  
    return "¡Hola, $nombre!";  
}
```

```
}
```

```
echo saludar("Alice"); // Imprime: ¡Hola, Alice!
```

```
?>
```

Conclusión

En los dos primeros capítulos de nuestro curso de PHP, hemos sentado las bases del desarrollo en este lenguaje. Presentamos PHP, exploramos su propósito, historia y cómo configurar un entorno de desarrollo local.

También aprendimos sobre la sintaxis básica, variables, tipos de datos, operadores y expresiones, conocimientos fundamentales para escribir scripts simples en PHP.

Además, analizamos estructuras de control como condicionales y bucles, lo que te permite crear programas dinámicos e interactivos. Finalmente, introducimos las funciones, una herramienta poderosa para organizar y reutilizar código de manera eficiente.