



## Guía Práctica de Ejercicios N° 1

En esta primer práctica estudiaremos los conceptos básicos relacionados con el lenguaje PHP. Comenzando con el funcionamiento de la Web, esquema de trabajo del lenguaje, historia y evolución de PHP, características, casos de éxito, requisitos para desarrollar con PHP, entre otros. Aprenderemos a embeber código PHP, comentar nuestros desarrollos, hacer uso de los tipos de variables, emplear variables, constantes, manejar operadores del lenguaje y estructuras de control del mismo.

**Objetivo:** Comprender el funcionamiento básico de la Web y PHP. También se pretende entender y emplear la sintaxis básica del lenguaje; luego lograr un manejo de las variables y constantes como así también de operadores y estructuras de control.

**Introducción:** En nuestra actualidad vemos que el desarrollo de Internet ha sido inminente y con ello las aplicaciones Web, por lo tanto, se hace indispensable el uso de un lenguaje que permita desarrollar aplicaciones Web como PHP, entre otros. Teniendo en cuenta la situación anterior es que veremos la como desarrollar aplicaciones Web que se ejecutarán del lado del servidor utilizando uno de los mejores lenguajes del ambiente del Software Libre.

### Construcciones (palabras clave) y funciones que estudiaremos en esta unidad.

#### Construcciones del lenguaje que veremos y editaremos.

**echo** nos permite dar salida a una o más cadenas pasadas como parámetros.

**print()** nos brinda la posibilidad de mostrar una cadena.

**if** nos permite la ejecución condicional de bloques de código PHP.

**else** la siguiente construcción extiende la sentencia **if** para ejecutar un bloque de código en caso de que la expresión de **if** se evalúe como falsa (**false**).

**elseif** es una combinación de **if** y **else**, permitiendo extender la condición del primer **if**.

**switch** es similar a un conjunto de sentencias **if** en la misma expresión.

**while** nos permite ejecutar un bloque de código repetidamente, mientras la expresión se evalúe como verdadera (**true**).

**do .. while** trabaja de igual manera que **while** excepto que la expresión se comprueba al final de cada iteración en vez de al principio.

**for** nos permite repetir un código dependiendo de un contador.

**foreach** nos brinda la posibilidad de iterar de forma sencilla matrices.

**break** permite escapar de una estructura de control iterante (bucle) como **for**, **while** o **switch**.

**continue** permite saltar el resto de la iteración actual del bucle y continuar la ejecución al



comienzo de la siguiente iteración.

### Funciones con las que trabajaremos.

**phpinfo()** nos brinda información sobre el interprete PHP instalado en nuestro servidor.

**gettype()** devuelve el tipo de una variable.

**is\_array()** devuelve true si la variable es un arreglo.

**is\_boolean()** retorna true si la variable es de tipo booleano.

**is\_int()** devuelve true si la variable es de tipo entero.

**is\_float()** retorna true si la variable es de tipo flotante.

**is\_object()** devuelve true si la variable es un objeto.

**is\_string()** retorna true si la variable es de tipo cadena.

**is\_resource()** devuelve true si la variable es un recurso.

**var\_dump()** permite mostrar el tipo y el valor de una variable.



## Ejercicios

### Preparar y verificar nuestro entorno de desarrollo PHP

Antes de comenzar con la práctica, asegúrese de tener instalado su entorno de desarrollo.

1) - Realice un script PHP que permita mostrar las características de interprete PHP utilizando la función `phpinfo()`.

```
<?php  
    phpinfo();  
?>
```

### Utilizar los conceptos básicos del lenguaje

2) - Realice un script que permita mostrar en pantalla el siguiente texto: Bienvenido al lenguaje PHP!!!. Recuerde utilizar etiquetas estándares y la construcción `echo`.

```
<?php  
    echo "Bienvenido al lenguaje PHP!!!";  
?>
```

3) - Edite el script anterior y agregue una nueva sentencia que muestre el siguiente texto: "PHP es un lenguaje de programación interpretado que se ejecuta del lado del servidor". Utilice la función `print()` para mostrar el texto.

```
<?php  
    echo "Bienvenido al lenguaje PHP!!!";  
    print("PHP es un lenguaje de programación interpretado  
        que se ejecuta del lado del servidor");  
?>
```

4) - Edite el script del punto 2 y agregue un comentario de línea a cada instrucción (`echo` y `print()`) indicando la función de cada una.



*// Permite enviar a la salida estándar el String especificado como parámetro*

*// La función print() trabaja de forma idéntica a la construcción echo*

### Utilizar y manejar los distintos tipos de variables y constantes

5) – Escriba un script PHP que permita declarar e inicializar variables de los siguientes tipos: **integer**, **float**, **string** y **boolean**. Luego muestre en pantalla el valor de cada una de las variables antes mencionadas.

```
<?php
    $variableEntera = 15;
    $variableFlotante = 1.7;
    $variableCadena = 'Variable de tipo Cadena';
    $variableBooleana = true;
    echo $variableEntera."<br/>";
    echo $variableFlotante."<br/>";
    echo $variableCadena."<br/>";
    echo $variableCadena."<br/>";
?>
```

6) – Editar el script PHP del punto anterior y verifique que cada variable es del tipo declarado utilizando las funciones para tal caso (**is\_int()**, **is\_float()**, **is\_string()**, etc.). Luego notifique por pantalla la verificación de cada una de las variables.

```
<?php
    $variableEntera = 15;
    $variableFlotante = 1.7;
    $variableCadena = 'Variable de tipo Cadena';
    $variableBooleana = true;
    if (is_int($variableEntera)) {
```



```
        echo "La variable \$variableEntera es de tipo int.";
    }
    if (is_float($variableFlotante)) {
        echo "La variable \$variableFlotante es de tipo float.";
    }
    if (is_string($variableCadena)) {
        echo "La variable \$variableCadena es de tipo string.";
    }
    if (is_bool($variableBooleana)) {
        echo "La variable \$variableBooleana es de tipo boolean.";
    }
}

?>
```

## Utilizar y manejar los operadores del lenguaje

7) – Escriba un script PHP que utilice los operadores matemáticos del lenguaje, realizando una suma, una resta, una multiplicación, una división y resto de la misma definiendo e inicializando dos variables para llevar a cabo las operaciones antes mencionadas. Deberá ir mostrando los resultados de cada operación en pantalla y también deberá documentar cada operación.

```
<?php
$a = 2;
$b = 1;
$suma = $a + $b;
echo "La suma de \$a y \$b es: ".$suma."<br/>";
$resta = $a - $b;
echo "La resta de \$a y \$b es: ".$resta."<br/>";
$multiplicacion = $a * $b;
echo "La multiplicación de \$a y \$b es: ".$multiplicacion."<br/>";
$division = $a / $b;
echo "La división de \$a y \$b es: ".$division."<br/>";
$resto = $a % $b;
```



```
echo "El resto de \$a y \$b es: ".$resto;
```

```
?>
```

8) – Escriba un script PHP que utilice los operadores de comparación del lenguaje, realizando una comparación de igualdad, de desigualdad (distinto), una comparación de menor, una de mayor, menor igual y mayor igual definiendo e inicializando dos variables para llevar a cabo las operaciones antes mencionadas. Deberá utilizar la sentencia if para determinar en cada caso el valor de verdad de la comparación, mostrando los resultados de cada operación en pantalla y también deberá documentar cada operación.

```
<?php
```

```
$a = 2;
```

```
$b = 1;
```

```
// Se realiza una comparación de igualdad
```

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

```
    echo "¡$a es igual a $b ". "<br/>";
```

```
}
```

```
// Se realiza una comparación de desigualdad
```

```
if ($a != $b) {
```

```
    echo "¡$a es distinto de $b". "<br/>";
```

```
}
```

```
// Se realiza una comparación de menor que
```

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

```
    echo "¡$a es menor que $b". "<br/>";
```

```
}
```

```
// Se realiza una comparación de mayor que
```

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

```
    echo "¡$a es mayor que $b". "<br/>";
```

```
}
```

```
// Se realiza una comparación de menor igual a
```

```
if ($a <= $b) {
```

```
    echo "¡$a es menor igual que $b". "<br/>";
```



```
}  
  
// Se realiza una comparación de mayor igual a  
if ($a >= $b) {  
    echo "¡$a es mayor igual que $b!". "<br/>";  
}  
  
?>
```

9) – Escriba un script PHP que utilice los operadores de incremento del lenguaje, realizando un incremento de una variable y decremento de la misma. Deberá ir mostrando los resultados de cada operación en pantalla y también deberá documentar cada operación.

```
<?php  
  
// Se inicializa la variable $a en cero  
$a = 0;  
  
echo "El valor de $a es: ".$a."<br/>";  
  
// Se realiza una operación de incremento  
$a++;  
  
echo "El valor de $a con incremento es: ".$a."<br/>";  
  
// Se realiza una operación de decremento  
$a--;  
  
echo "El valor de $a con decremento es: ".$a."<br/>";  
  
?>
```

10) – Escriba un script PHP que utilice los operadores lógicos del lenguaje, realizando un comparativas del operador **and (&&)**, **or (||)**, **xor** y **not (!)** definiendo e inicializando dos variables para llevar a cabo las operaciones antes mencionadas. Deberá ir mostrando los resultados de cada operación en pantalla y también deberá documentar cada operación.

```
<?php  
  
$a = 2;  
$b = 1;  
  
// Se realiza una comparación de conjunción and
```



```
if ($a > $b && $a == 2) {  
    echo "¡$a es mayor que $b y es igual a 2!"<br/>";  
}  
if ($a > $b || $a == 2) {  
    echo "¡$a es mayor que $b o es igual a 2 o ambas!"<br/>";  
}  
if ($a > $b xor $a == 2) {  
    echo "¡$a es mayor que $b o es igual a 2 pero  
    no ambas!"<br/>";  
}  
if (!($b > $a && $b == 2)) {  
    echo "¡$b es no mayor que $a y es no igual a 2!"<br/>";  
}  
?>
```

## Utilizar y manejar las estructuras de control del lenguaje

11) – Realice un script PHP que utilice la construcción **if** para determinar en que estación (verano, otoño, invierno y primavera) del año estamos de acuerdo al número de mes que se va a definir e inicializar una variable de nombre mes para llevar a cabo las operaciones de comparación. Deberá ir mostrando los resultados de cada operación en pantalla y también deberá documentar cada operación.

```
<?php  
$mes = 4;  
    // Consulta los meses del año correspondientes a cada Estación  
if ($mes == 1 || $mes == 2) {  
    echo "Estamos en Verano";  
}  
if ($mes == 4 || $mes == 5) {  
    echo "Estamos en Otoño";  
}
```





```
if ($mes == 7 || $mes == 8) {  
    echo "Estamos en Invierno";  
}  
if ($mes == 10 || $mes == 11) {  
    echo "Estamos en Primavera";  
}  
if ($mes == 3) {  
    echo "Estamos en Verano u Otoño";  
}  
if ($mes == 6) {  
    echo "Estamos en Otoño o Invierno";  
}  
if ($mes == 9) {  
    echo "Estamos en Invierno o Primavera";  
}  
if ($mes == 12) {  
    echo "Estamos en Primavera o Verano";  
}  
?  
>
```

**12)** – Escriba un nuevo script PHP que realice la misma tarea del punto anterior pero utilizando la construcción **if – elseif**. Deberá ir mostrando los resultados de cada operación en pantalla y también deberá documentar cada operación.

```
<?php  
$mes = 4;  
  
// Se consulta por los meses del año correspondientes a cada Estación  
if ($mes == 1 || $mes == 2) {  
    echo "Estamos en Verano";  
} else if ($mes == 4 || $mes == 5) {  
    echo "Estamos en Otoño";  
}
```



```
    } else if ($mes == 7 || $mes == 8) {  
        echo "Estamos en Invierno";  
    } else if ($mes == 10 || $mes == 11) {  
        echo "Estamos en Primavera";  
    } else if ($mes == 3) {  
        echo "Estamos en Verano u Otoño";  
    } else if ($mes == 6) {  
        echo "Estamos en Otoño o Invierno";  
    } else if ($mes == 9) {  
        echo "Estamos en Invierno o Primavera";  
    } else if ($mes == 12) {  
        echo "Estamos en Primavera o Verano";  
    }  
    ?>
```

13) – Realice un nuevo script PHP que permita determinar el día de la semana utilizando la estructura **switch** para tal fin, deberá declarar e inicializar una variable de tipo entera para guardar el número de día de la semana que comienza con el día Domingo y termina con el día Sábado (Domingo = 0, Lunes = 1, Martes = 2, etc.). Deberá ir mostrando los resultados de cada operación en pantalla y también deberá documentar cada operación.

```
<?php  
$mes = 4;  
  
    // Se consulta por los meses del año correspondientes a cada  
    Estación  
    switch ($mes) {  
        case 1:  
            echo "Estamos en Verano";  
            break;  
        case 2:  
            echo "Estamos en Verano";  
            break;
```



```
case 3:
    echo "Estamos en Verano u Otoño";
    break;

case 4:
    echo "Estamos en Otoño";
    break;

case 5:
    echo "Estamos en Otoño";
    break;

case 6:
    echo "Estamos en Otoño o Invierno";
    break;

case 7:
    echo "Estamos en Invierno";
    break;

case 8:
    echo "Estamos en Invierno";
    break;

case 9:
    echo "Estamos en Invierno o Primavera";
    break;

case 10:
    echo "Estamos en Primavera";
    break;

case 11:
    echo "Estamos en Primavera";
    break;

case 12:
    echo "Estamos en Primavera o Verano";
    break;
```



*default:*

*echo "Número de mes incorrecto.";*

*}*

*?>*

14) – Realice un nuevo script PHP que recorra números del 1 al 100 y que calcule el doble de cada uno de ellos. Utilice la sentencia **while** para recorrer los mismos. Deberá ir mostrando los resultados de cada operación en pantalla y también deberá documentar cada operación.

*<?php*

*\$c = 1;*

*// Se recorren los números que van del 1 al 100 con while*

*while (\$c <= 100) {*

*echo (\$c\*2). "<br/>";*

*\$c++;*

*}*

*?>*

15) – Escriba un nuevo script PHP que realice la misma tarea del punto anterior pero que utilice la sentencia **do .. while** para recorrer los mismos. Deberá ir mostrando los resultados de cada operación en pantalla y también deberá documentar cada operación.

*<?php*

*\$c = 1;*

*// Se recorren los números que van del 1 al 100 con do .. while*

*do {*

*echo (\$c\*2). "<br/>";*

*\$c++;*

*} while (\$c <= 100);*

*?>*



**16)** – Para terminar con la práctica de esta clase realice un nuevo script PHP que recorra números del 1 al 100 y que calcule el cuadrado de cada uno de ellos. Utilice la sentencia **for** para recorrer los mismos. Deberá ir mostrando los resultados de cada operación en pantalla y también deberá documentar cada operación.

```
<?php  
    for ($c = 1; $c <= 100; $c++) {  
        echo ($c*$c)."<br/>";  
    }  
?>
```