





Funciones

```
<?php
    function ejemplo($arg1, $arg2, ... , $argN = null) {
        echo "Función de ejemplo.\n";
        return $valor_devuelto;
    }
?>
```



Funciones

```
<?php
    function sumar($num1, $num2) {
        return $num1 + $num2;
    }

    echo sumar(82, 98);

?>
```



Funciones

```
<?php
```

```
    function saludar($nombre, $apellido = "") {  
        echo $nombre . $apellido;  
    }
```

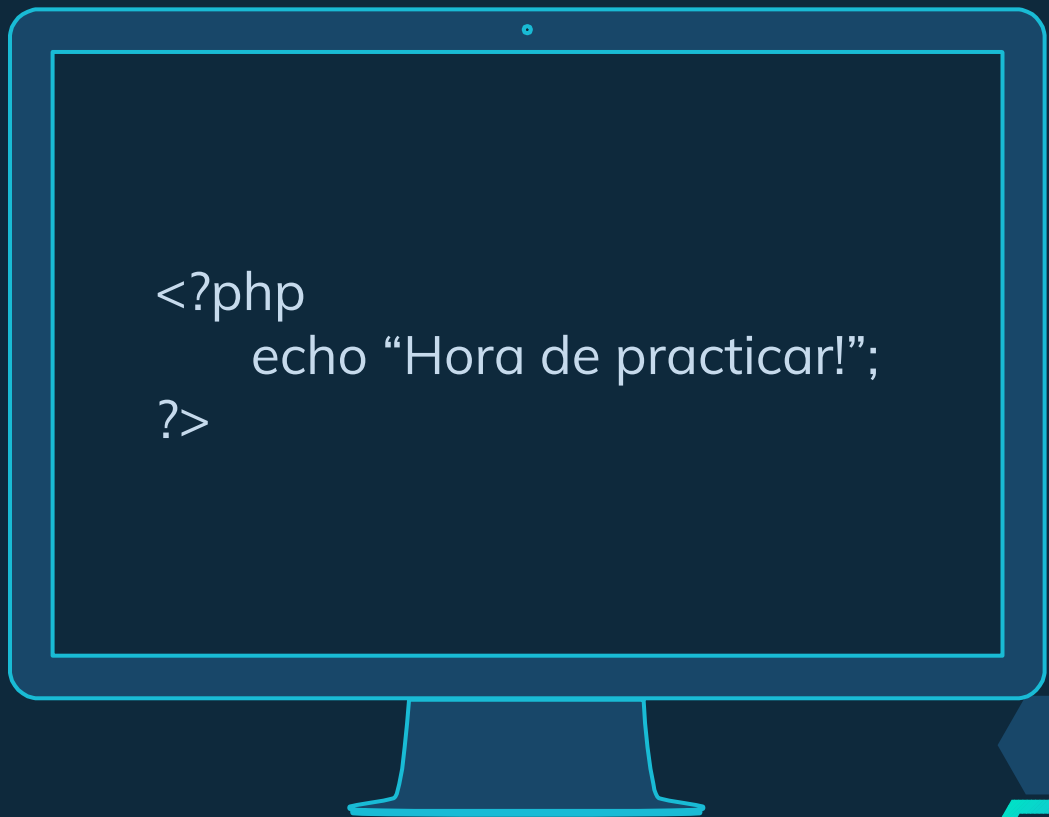
```
    $saludo = saludar("Juan", "Perez");  
    $saludo2 = saludar("Ricardo");
```

```
?>
```



¡A practicar!

Ejercicios del 1 al 2





Scope

El ámbito de una variable es el contexto dentro del que la variable está definida. La mayor parte de las variables PHP, sólo tienen un ámbito simple.

PHP tiene 2 alcances posibles:

- ◇ Global
- ◇ Local





Scope

```
<?php  
    $a = 1; //ámbito global
```

```
function test() {  
    echo $a; //ámbito local  
}  
test();
```

```
?>
```

No produce salida!






Scope

```
<?php  
    $a = 1; //ámbito global
```

```
function test() {  
    global $a;  
    echo $a;  
}  
test();  
?>
```





Static

```
<?php
    function test() {
        $a = 0;
        echo ++$a;
    }
    test();
    test();
    test();
?>
```





Static

```
<?php
    function test() {
        static $a = 0;
        echo ++$a;
    }
    test();
    test();
    test();
?>
```





Inclusión de Archivos

- ◇ include
- ◇ include_once
- ◇ require
- ◇ require_once

```
<?php  
    require("soporte.php");  
?>
```





¿Cómo hacemos Debug?



echo

var_dump

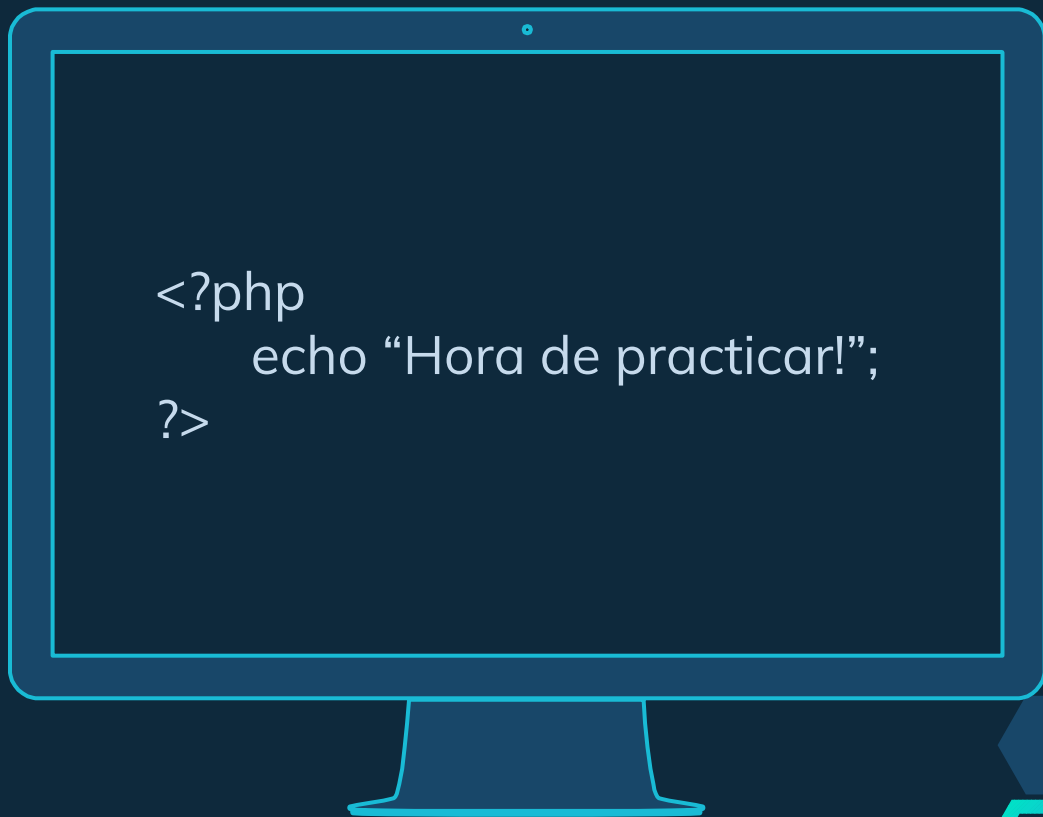
exit





¡A practicar!

Ejercicios del 3 al 7





Scripting

Y teniendo todos estos recursos...

¿para que nos puede servir PHP?



Scripting

Y teniendo todos estos recursos...

¿para que nos puede servir PHP?

- Para hacer sitios web.



Scripting

Y teniendo todos estos recursos...

¿para que nos puede servir PHP?

- Para hacer sitios web.
- ¡Para procesar información en otros formatos!



Scripting

Para ejecutar procesos internos muchas veces no utilizamos un navegador ni como punto de entrada ni como punto de salida. Entre otras cosas podemos usar una **consola**.

Para esto utilizaremos el comando **php**



Scripting

php -r nos permite ejecutar código definido en la misma línea:

```
php -r 'echo "Hola Mundo";'
```



Scripting

php -a nos permite acceder al modo interactivo

php -f nos permite ejecutar un archivo:

php -f scripting.php



readline()

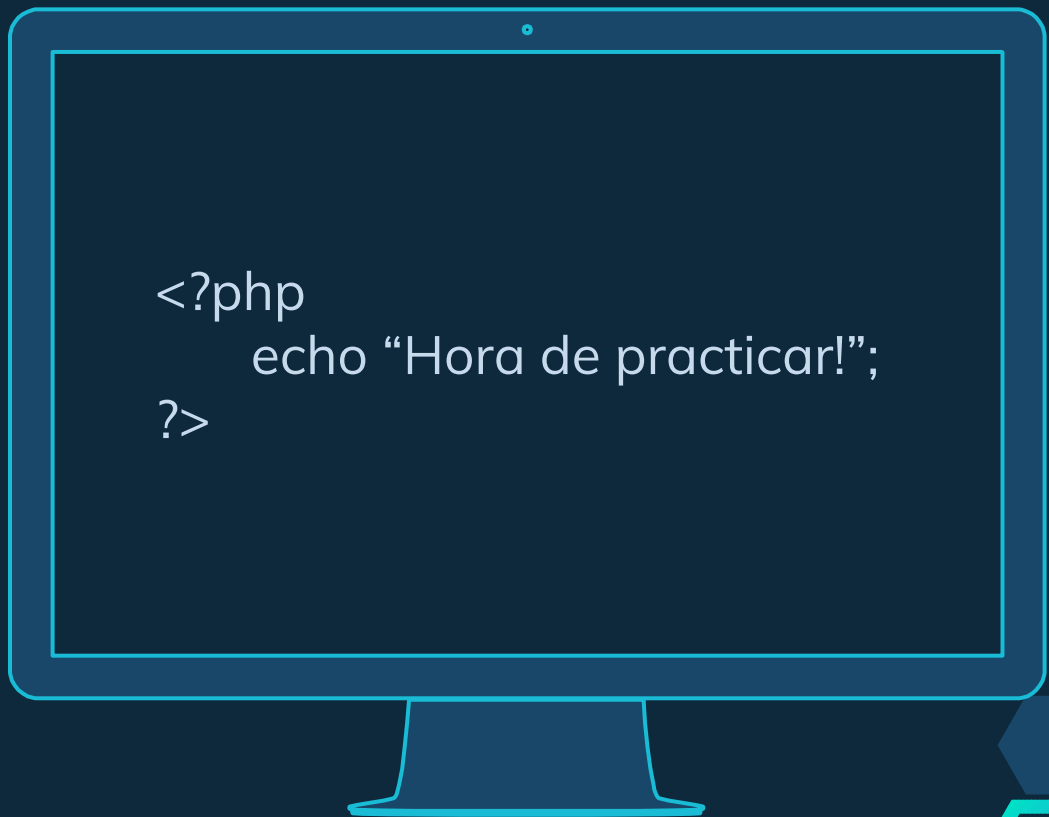
La función **readline()** nos permite interactuar con el usuario desde la terminal para pedirle información

Uso:

```
$nombre = readline("Ingrese su nombre: ");
```



¡A practicar!





Gracias!

¿Preguntas?

