Contents

[**Comillas** 1](#_Toc56747465)

[**Extracción de datos** 3](#_Toc56747466)

[**Formato de datos** 6](#_Toc56747467)

[**Expresiones regulares** 8](#_Toc56747468)

[**Funciones: Argumentos** 9](#_Toc56747469)

[**Return** 12](#_Toc56747470)

[**Closure** 13](#_Toc56747471)

[**Array simple** 13](#_Toc56747472)

[**Array complejo** 14](#_Toc56747473)

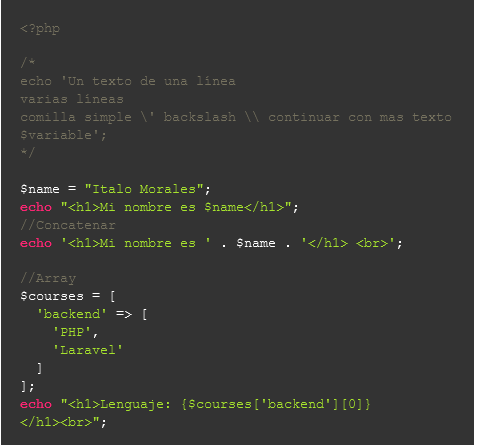
[**Funciones PHP para arrays** 16](#_Toc56747474)

[**Comparación** 21](#_Toc56747475)

[**Unión** 22](#_Toc56747476)

# **Comillas**

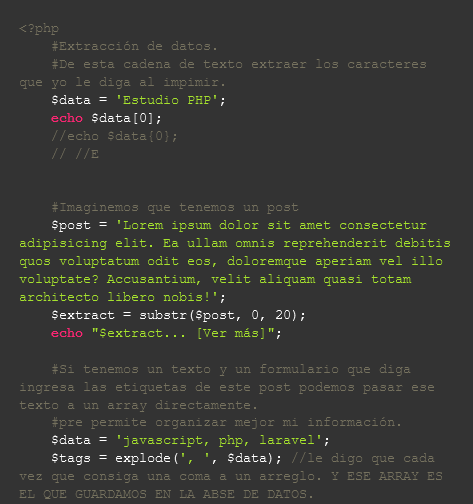
* **Simples**: usadas para textos. los caracteres. Estas no pueden obtener datos de una variable, la salida ser texto.
* **Dobles**: Extienden a las variables, pudiendo agregarlas a la cadena y estas serán reconocidas.



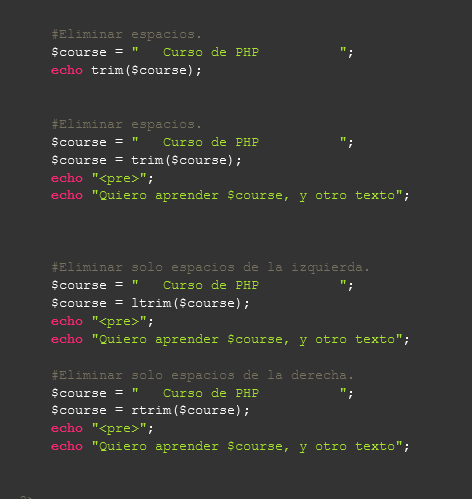




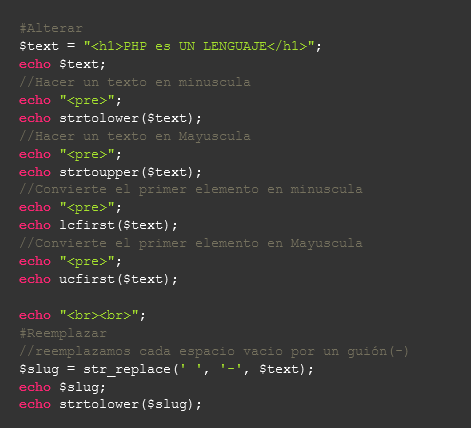
# **Extracción de datos**

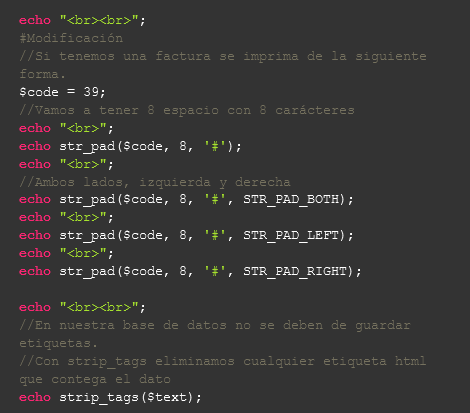


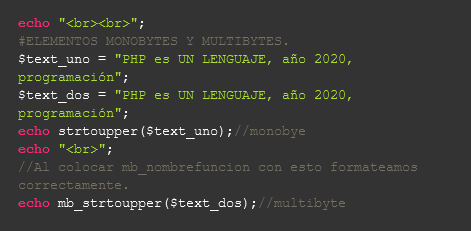




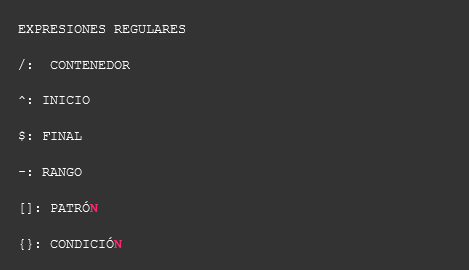
# **Formato de datos**

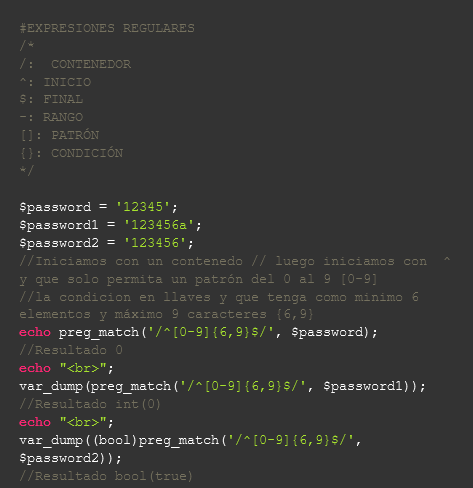






# **Expresiones regulares**

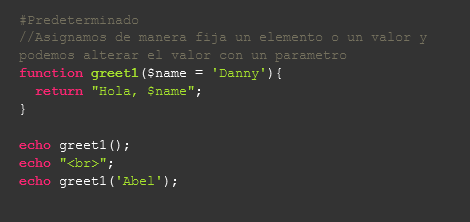




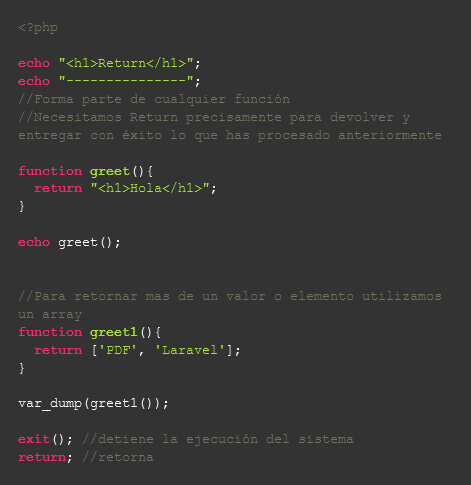
# **Funciones: Argumentos**

* **Argumentos**: Colocamos dentro de los paréntesis, un valor o un parámetro.
* **Referencia**: Apunta u observa el comportamiento de otro elemento.
* **Valores**: Colocas una variable directamente esperando a que se cumpla.
* **Referencia**: Además de hacer un cambio interno, también afecte al elemento que este fuera.





# **Return**

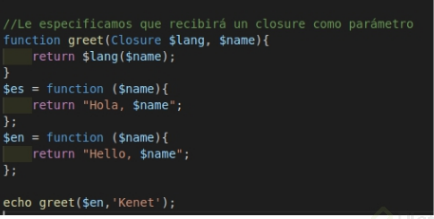


**return** devuelve el resultado de toda la operación de la función, por tanto, cuando igualas una variable a una función, en la variable se va a guardar lo que la función retorne.

# **Closure**

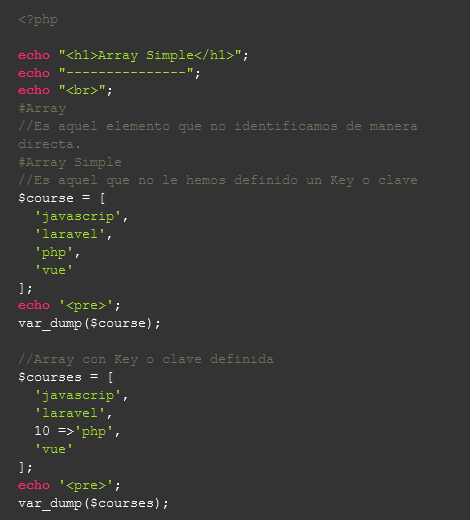
También conocidas como **funciones anónimas**.

* Se usa cuando una variable requiere lógica.
* La palabra reservada **Closure** me permite especificar, forzar el tipo de parámetro que le debe llegar.



# **Array simple**

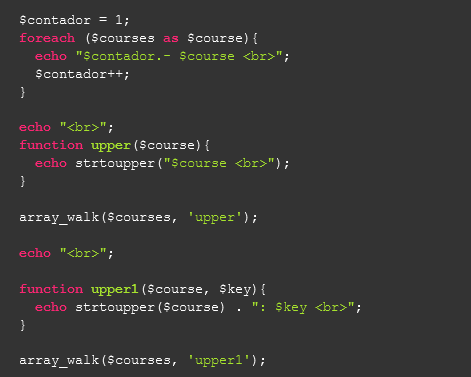
Elemento al cual no se le ha definido una **key**. Los valores no se identifican de manera directa.



# **Array complejo**

Podemos pasar de **simple** a **complejo** determinando los keys de cada elemento del array, es decir, hacer **arrays asociativos**.

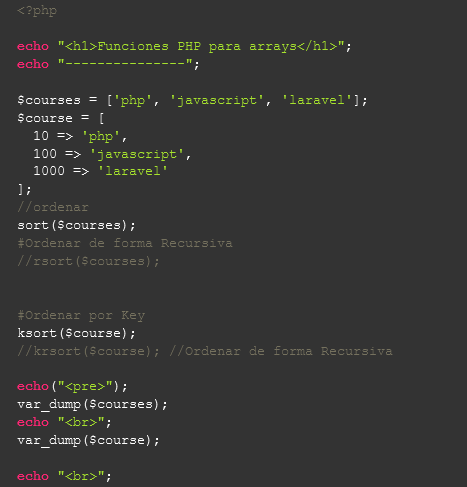


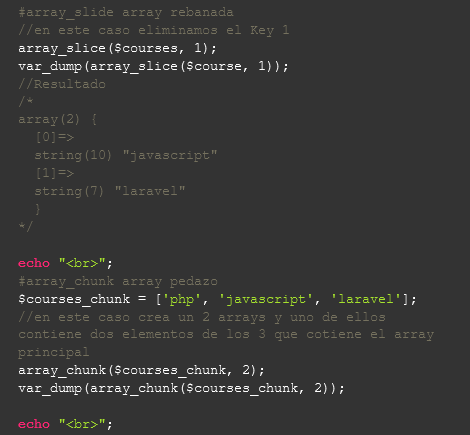


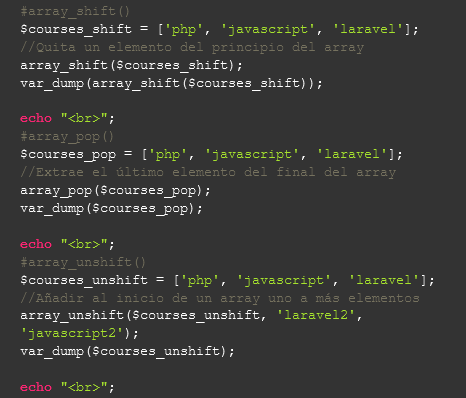
# **Funciones PHP para arrays**

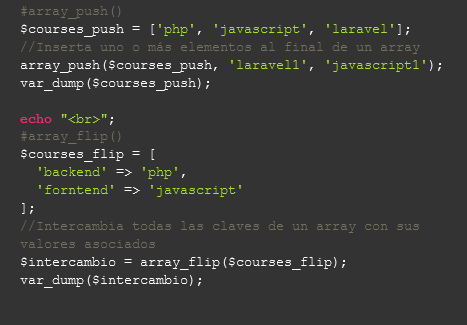
Estas son algunas de las funciones de PHP utilizadas para los arrays:

* **sort**: Ordena un array.
* **rsort**: Ordena un array en orden inverso.
* **ksort**: Ordena un array por clave.
* **krsort**: Ordena un array por clave en orden inverso.
* **array\_slice**: Extrae una parte de un array.
* **array\_chunk**: Divide un array en fragmentos.
* **array\_shift**: Quita un elemento del principio del array.
* **array\_pop**: Extrae el último elemento del final del array.
* **array\_unshift**: Añadir al inicio de un array uno a más elementos.
* **array\_push**: Inserta uno o más elementos al final de un array.
* **array\_flip**: Intercambia todas las claves de un array con sus valores asociados.

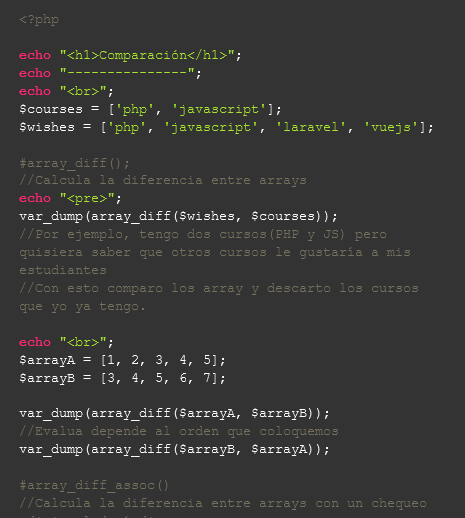








# **Comparación**



# **Unión**



