

# PHP

## Sintaxis básica:

`<?php ¿>` Etiqueta de apertura y cierre.

`echo ' ';` Muestra los datos dentro de las comillas.

## Constantes:

`Const name= ' ';`

`define ('name', '')`

## Variables:

`$variable= ' ';` Definir variable.

## Variables predefinidas:

`$GLOBALS[ ' ']` Se utiliza para acceder a las variables globales desde cualquier script.

`$_SERVER[ ' ']` Contiene información sobre la ubicación de rutas.

`$_GET[ ' ']` Puede recopilar datos que se enviaron en la url.

`$_POST[ ' ']` Se utiliza para recopilar datos de un formulario.

`$_REQUEST[ ' ']` Se utiliza para recopilar datos de un formulario.

## Array:

`$variable= array();` Definir array.

`$variable= [ ];` Definir array.

`$Variable[ ' '];` Llamar array.

## Funciones para manejo de variables:

**Boolval();** Se utiliza para recuperar valores booleano.

**empty();** Comprueba si una variable esta vacía o no.

**floatval();** Obtener valores flotantes de una variable.

**import\_request\_variables();** Importa variables GET/POST/Cookie en el ámbito local.

**intval();** Encuentra el valor entero de una variable.

**is\_array();** Comprueba si es una array.

**is\_bool();** Comprueba si es un valor booleano.

**is\_countable();** Verifica si el contenido de una variable es contable.

**is\_float();** Verifica si es un valor float.

**is\_int();** Verifica si es un numero.

**is\_iterable();** Verifica si es un valor nulo

**is\_null();** Verifica si es un valor nulo.

**is\_numeric();** Verifica si es un valor numérico.

**is\_object();** Verifica si es un objeto.

**is\_string();** Verifica si es un string.

**isset();** Verifica si existe y no es null.

**unset();** Desinstala una variable.

**var\_dump();** Proporciona información sobre una variable.

## Includes :

**Include 'ubicación relativa';** Incluye todo el código de un fichero.

**Require 'ubicacionrelativa';** Incluye todo el código de un fichero.

**Includeonce 'ubicación relativa';** Incluye todo el código de un fichero, no se puede repetir.

**Requireonce 'ubicacionrelativa';** Incluye todo el código de un fichero, no se puede repetir.

## Operadores aritméticos:

**+** Adición.

**-** Resta.

**\*** Multiplicación.

**/** División.

**%** Modulo.

**\*\*** Exponenciación.

## Operadores de comparación:

**==** Igual.

**!=** No es igual.

**<** Menos que.

**>** Mayor que.

**<=** Menor igual que.

**>=** Mayor igual que.

## Operadores lógicos:

**and** Y.

**or** O.

**!** No.

**&&** Y

**||** O

## Operadores de incremento:

**\$variable++** incremento +1

**\$variable--** decremento -1

## Bucles:

`while(condición){código}` Es el bucle más básico en php.

`for(variable contador, condición, incremento){código}`

Este bucle lo usamos cuando sabemos la cantidad exacta de repeticiones.

`foreach(variablearray as variableausar){código}` Este bucle funciona solo con array y objetos.

## Estructura de control condicional:

`if(){instrucciones}` Ejecuta el código si una condición es verdadera.

`else{instrucciones}` Ejecuta el código si las condiciones no son verdaderas.

`elseif{instrucciones}` Ejecuta diferentes condiciones.

## Funciones:

`function Name(parametro) {código}`

`global $variable= '';`

`return $variable;`

## Sesiones:

`session_star();`

`$_session[''];`

`session_destroy('session a destruir')`

`unset('session a vaciar')`

## Cookies:

`setcookie('nombre', 'valor', 'caducidad')` Crear cookie.

`$_cookie['nombre']` Mostar cookie.

## Constantes de filtro:

**FILTER\_VALIDATE\_BOOLEAN** Valida un booleano.

**FILTER\_VALIDATE\_EMAIL** Certifica una dirección de correo electrónico.

**FILTER\_VALIDATE\_FLOAT** Confirma un flotador.

**FILTER\_VALIDATE\_INT** Verifica un número entero.

**FILTER\_VALIDATE\_IP** Valida una dirección IP.

**FILTER\_VALIDATE\_URL** Valida una URL.

## Funciones de mysql:

**mysqli\_affected\_rows()** El número de filas afectadas en la operación anterior de MySQL.

**mysqli\_autocommit()** Activar o desactivar las modificaciones de la base de datos de confirmación automática.

**mysqli\_change\_user()** Cambia el usuario de la conexión de base de datos especificada.

**mysqli\_character\_set\_name()** El juego de caracteres predeterminado para la conexión a la base de datos.

**mysqli\_close()** Cierra una conexión de base de datos abierta.

**mysqli\_commit()** Confirma la transacción actual.

**mysqli\_connect\_errno()** El código de error del último error de conexión.

**mysqli\_connect()** Abre una nueva conexión al servidor MySQL.

**mysqli\_debug()** Realiza operaciones de depuración.

**mysqli\_dump\_debug\_info()** Volca la información de depuración en un registro.

**mysqli\_errno()** El último código de error para la llamada de función más reciente.

**mysqli\_error\_list()** Una lista de errores para la llamada de función más reciente.

**mysqli\_error()** La última descripción de error para la llamada de función más reciente.

**mysqli\_fetch\_all()** Obtiene todas las filas de resultados como una matriz.

**mysqli\_fetch\_array()** Obtiene una fila de resultados como asociativa, matriz numérica o ambas.

**mysqli\_fetch\_assoc()** Obtiene una fila de resultados como una matriz asociativa.

**mysqli\_fetch\_field\_direct()** Metadatos para un solo campo como objeto.

**mysqli\_fetch\_field()** El siguiente campo en el conjunto de resultados como un objeto.

**mysqli\_fetch\_fields()** Una matriz de objetos que representan los campos en un conjunto de resultados.

**mysqli\_fetch\_lengths()** Las longitudes de las columnas de la fila actual en el conjunto de resultados.

**mysqli\_fetch\_object()** La fila actual de un conjunto de resultados como objeto.

**mysqli\_fetch\_row()** Obtiene una fila de un conjunto de resultados y la devuelve como una matriz enumerada.

**mysqli\_field\_count()** El número de columnas de la consulta más reciente.

**mysqli\_field\_seek()** Establece el cursor de campo en el desplazamiento de campo dado.

**mysqli\_field\_tell()** La posición del cursor de campo.

**mysqli\_free\_result()** Libera la memoria asociada a un resultado.

**mysqli\_get\_charset()** Un objeto de juego de caracteres.

**mysqli\_get\_client\_info()** La versión de la biblioteca cliente MySQL.

**mysqli\_get\_client\_stats()** Devuelve las estadísticas del cliente por proceso.

**mysqli\_get\_client\_version()** La versión de la biblioteca cliente MySQL como un número entero.

**mysqli\_get\_connection\_stats()** Estadísticas sobre la conexión del cliente.

**mysqli\_get\_host\_info()** El nombre de host del servidor MySQL y el tipo de conexión.

**mysqli\_get\_proto\_info()** La versión del protocolo MySQL.

**mysqli\_get\_server\_info()** Devuelve la versión del servidor MySQL.

**mysqli\_get\_server\_version()** La versión del servidor MySQL como un número entero

**mysqli\_info()** Devuelve información sobre la consulta ejecutada más recientemente.

**mysqli\_init()** Inicializa MySQLi y devuelve un recurso para usar con mysqli\_real\_connect().

**mysqli\_insert\_id()** Devuelve el ID generado automáticamente utilizado en la última consulta.

**mysqli\_kill()** Pide al servidor que elimine un hilo de MySQL.

**mysqli\_more\_results()** Comprueba si hay más resultados de una consulta múltiple.

**mysqli\_multi\_query()** Realiza una o más consultas en la base de datos.

**mysqli\_next\_result()** Prepara el siguiente conjunto de resultados de `mysqli_multi_query()`.

**mysqli\_num\_fields()** El número de campos en un conjunto de resultados.

**mysqli\_num\_rows()** El número de filas en un conjunto de resultados.

**mysqli\_options()** Establece opciones de conexión adicionales y afecta el comportamiento de una conexión.

**mysqli\_ping()** Hace ping a la conexión del servidor o intenta volver a conectarse si no funciona.

**mysqli\_prepare()** Prepara una declaración SQL para su ejecución.

**mysqli\_query()** Realiza una consulta contra la base de datos.

**mysqli\_real\_connect()** Abre una nueva conexión al servidor MySQL.

**mysqli\_real\_escape\_string()** Escapa de caracteres especiales en una cadena para usar en una declaración SQL.

**mysqli\_real\_query()** Ejecuta una consulta SQL.

**mysqli\_reap\_async\_query()** Devuelve el resultado de una consulta asincrónica.

**mysqli\_refresh()** Actualiza tablas o cachés o restablece la información del servidor de replicación.

**mysqli\_rollback()** Deshace la transacción actual para la base de datos.

**mysqli\_select\_db()** Cambia la base de datos predeterminada para la conexión.

**mysqli\_ssl\_set()** Establece conexiones seguras usando SSL.

**mysqli\_stat()** El estado actual del sistema.

**mysqli\_stmt\_init()** Inicializa una declaración y devuelve un objeto para usar con `mysqli_stmt_prepare()`.

**mysqli\_store\_result()** Transfiere un conjunto de resultados de la última consulta.

**mysqli\_thread\_id()** El ID del hilo de la conexión actual.

**mysqli\_thread\_safe()** Devuelve si la biblioteca cliente está compilada como segura para subprocesos.

**mysqli\_use\_result()** Inicia la recuperación de un conjunto de resultados de la última consulta ejecutada usando el `mysqli_real_query()`.

**mysqli\_warning\_count()** El número de advertencias de la última consulta en la conexión.

## Funciones de fecha / hora

**date\_add()** Agrega una cantidad de años hasta segundossegundos a un objeto de fecha.

**date\_create\_from\_format()** Devuelve un objeto DateTime formateado.

**date\_create()** Crea un nuevo objeto DateTime.

**date\_date\_set()** Establece una nueva fecha.

**date\_default\_timezone\_get()** Devuelve la zona horaria predeterminada utilizada por todas las funciones.

**date\_default\_timezone\_set()** Establece la zona horaria predeterminada.

**date\_diff()** Calcula la diferencia entre dos fechas.

**date\_format()** Devuelve una fecha formateada según un formato específico.

**date\_interval\_format()** Formatea un intervalo.

**date\_isodate\_set()** Establece una fecha de acuerdo con las normas ISO 8601.

**date\_modify()** Modifica la marca de tiempo.

**date\_parse()** Devuelve una matriz con información detallada sobre una fecha.

**date\_sub()** Resta días, meses, años, horas, minutos y segundos de una fecha.

**date\_time\_set()** Establece la hora.

**date\_timestamp\_get()** Devuelve la marca de tiempo de Unix.

**date\_timestamp\_set()** Establece la fecha y la hora según marca de tiempo de Unix.

**date\_timezone\_get()** Devuelve la zona horaria de un objeto DateTime dado.

**date\_timezone\_set()** Establece la zona horaria para un objeto DateTime.

**date()** Formatea una fecha y hora local.

**getdate()** Información de fecha/hora de tiempo o la fecha/hora local actual.

**gettimeofday()** La hora actual.

**gmdate()** Formatea una fecha y hora GMT / UTC.

**gmmktime()** La marca de tiempo de Unix para una fecha GMT.

**gmstrftime()** Formatea una fecha y hora GMT / UTC de acuerdo con la configuración regional.

**localtime()** la hora local.

**mktime()** La marca de tiempo de Unix para una fecha.

**time()** La hora actual como marca de tiempo de Unix.