

JAVASCRIPT	PHP
INICIO	
(En HTML) => <code><script> -código- </script></code>	Tanto en HTML como en PHP => <code><?php -código- ?></code> <code><? echo -código- ?> ➔ <?= -código- ?></code>
ACCESO	
<code><script src="ruta/archivo.js"></script></code>	<code><form action="ruta/archivo.php"></form></code>
DECLARACIÓN DE VARIABLES	
<code>var / let variable = valor;</code>	<code>\$variable = valor;</code>
TIPO DE VARIABLE	
<code>typeof(variable);</code>	<code>gettype(\$variable);</code>
ARRAYS	
<code>let nombre_array = [val1, val2, val3,];</code>	<code>\$nombre_array = array(val1, val2, val3,);</code>
<code>let primerValor = nombre_array[0]; // = "val1";</code>	<code>\$primerValor = \$nombre_array[0]; // = "val1";</code>
DICIONARIOS	
<code>let diccionario = { "clave1":valor1, "clave2":valor2,...};</code>	<code>\$diccionario = array("clave1"=>valor1, "clave2"=>valor2,...);</code>
BUCLES	
<code>for (let index = 0; index < valorMáximo; index++){ -código- }</code>	<code>for (\$index = 0; \$index < valorMáximo; \$index++){ -código- }</code>
<code>for (let index = valorMáximo; index > 0; index--){ -código- }</code>	<code>for (\$index = valorMáximo; \$index > 0; \$index --){ -código- }</code>
<code>array.foreach((elemento) => { -código- });</code>	<code>foreach (\$array as \$valor) { -código- }</code>
<code>while (condición) { -código- }</code>	<code>=</code>
MÉTODOS PARA STRINGS	
<code>let cadena = "*****";</code>	<code>\$cadena = "*****";</code>
<code>let longitud = cadena.length;</code>	<code>\$longitud = strlen(\$cadena);</code>
<code>let aMinúsculas = cadena.toLowerCase();</code>	<code>\$aMinusculas = strtolower(\$cadena);</code>
<code>let aMayusculas = cadena.toUpperCase();</code>	<code>\$aMayusculas = strtoupper(\$cadena);</code>
<code>let letra = cadena.charAt(indice);</code>	<code>\$letra = \$cadena[indice];</code>
<code>let arrayPalabras/Letras = cadena.split(separador);</code>	<code>\$arrayPalabras = explode(separador, \$cadena);</code> <code>\$arrayLetras = str_split(\$cadena);</code>
<code>let porcion = cadena.substring(posición inicio, posición final);</code>	<code>\$porcion = substr(\$cadena, posición inicio, posición final);</code>
<code>let nuevaCadena = cadena.replace(expresión, nuevoValor);</code>	<code>\$nuevaCadena = str_replace(\$cadena, expresión, nuevoValor);</code>
MÉTODOS PARA ARRAYS	
<code>let array = [val1, val2, val3];</code>	<code>\$array = [val1, val2, val3];</code>
<code>let longitud = array.length;</code>	<code>\$longitud = count(\$array);</code>
<code>let creaCadena = array.join(separador);</code>	<code>\$creaCadena = implode(separador, \$array);</code>
<code>let eliminaUltimoElemento = array.pop();</code>	<code>\$eliminaUltimoElemento = array_pop(\$array);</code>
<code>let eliminaPrimerElemento = array.shift();</code>	<code>\$eliminaPrimerElemento = array_shift(\$array);</code>
<code>array.push(nuevoElemento); // Añade elemento al final</code>	<code>array_push(\$array, elemento);</code>
<code>array.unshift(nuevoElemento); // Añade elemento al principio</code>	<code>array_unshift(\$array, elemento);</code>
<code>array.reverse(); // Coloca los elementos al inverso</code>	<code>\$arrayInvertido = array_reverse(\$array);</code>
FUNCIONES	
<code>function miFuncion(parametro1, parametro2, ...) { -codigo- }</code>	

JAVASCRIPT - DOM

OBTENER ELEMENTOS

document	Objeto javascript que hace referencia al contenido total de nuestra página.
document.getElementsByTagName(nombreEtiqueta)	Obtiene todos los elementos de la página por etiqueta/tag.
document.getElementsByName(nombreAtributo)	Obtiene todos los elementos de la página por el atributo NAME.
document.getElementById(nombreAtributo)	Obtiene todos los elementos de la página por el atributo ID.

ACCESO A ATRIBUTOS Y ESTILOS

<code>Enlace</code> <code>let enlace = document.getElementById("enlace");</code> <code>alert(enlace.href); // muestra http://www...com</code>	<code></code> <code>let imagen = document.getElementById("imagen");</code> <code>alert(imagen.style.margin);</code>
--	---

PÁGINA

window.location = "http://www.paginanueva.com";	Cambia la dirección de la página (en la misma ventana).
window.open ("pagina.html", "nombrepagina", "width=500,height=200");	Abre la dirección en una nueva ventana.

CREAR ELEMENTOS

<code>let parrafo = document.createElement("p");</code>	Creación de un nodo de tipo Element/Tag que represente al elemento.
<code>let atributo = document.createAttribute("id");</code>	Creación de un nodo de tipo Atributo .
<code>atributo.value = "nombre_id";</code>	Se define el contenido del atributo.
<code>let contenido = document.createTextNode("Hola Mundo!");</code>	Creación de un nodo de tipo Texto que represente el contenido del elemento.
<code>parrafo.appendChild(contenido);</code>	Añadir el nodo Text como nodo hijo del nodo Element.
<code>document.body.appendChild(parrafo);</code>	Añadir el nodo Element a la página, en forma de nodo hijo del nodo correspondiente al lugar en el que se quiere insertar el elemento.

ELIMINACIÓN DE ELEMENTOS

<code>var parrafo = document.getElementById("provisional");</code> <code>parrafo.parentNode.removeChild(parrafo);</code> <code><p id="provisional">...</p></code>	La forma más segura y rápida de acceder al nodo padre de un elemento es mediante la propiedad nodoHijo.parentNode .
---	--

JAVASCRIPT - EVENTOS	
ELEMENTOS	
ONBLUR	El elemento pierde el focus / la selección.
ONFOCUS	El elemento recibe el focus / es seleccionado.
ONCHANGE	Cuando el elemento detecta un cambio.
ONCLICK	Clicar con el ratón.
ONDBLCLICK	Clicar dos veces con el ratón.
ONSELECT	Al seleccionar un texto.
TECLAS	
ONKEYDOWN	Mantener pulsada una tecla.
ONKEYPRESS	Pulsar una tecla.
ONKEYUP	Soltar una tecla pulsada.
PÁGINA	
ONLOAD	La página se ha cargado completamente.
ONUNLOAD	Al abandonar la página / cerrar el navegador.
ONRESIZE	Al modificar el tamaño del navegador.
RATÓN / MOUSE	
ONMOUSEDOWN	Mantener pulsado un botón del ratón.
ONMOUSEUP	Al soltar el botón pulsado del ratón.
ONMOUSEMOVE	Al mover el ratón.
ONMOUSEOUT	El ratón sale del elemento.
ONMOUSEOVER	El ratón pasa por encima de un elemento.
FORMULARIO	
ONRESET	Borra los datos del formulario.
ONSUBMIT	Cuando se envía el formulario.

JAVASCRIPT
<p>THIS => Hace referencia al elemento que ha provocado el evento.</p> <pre> <div id="contenidos" style="width:150px; height:60px; border:thin solid silver" onmouseover="document.getElementById('contenidos').style.borderColor='black';" onmouseout="document.getElementById('contenidos').style.borderColor='silver';"> Sección de contenidos... </div> <div id="contenidos" style="width:150px; height:60px; border:thin solid silver" onmouseover="this.style.borderColor='black';" onmouseout="this.style.borderColor='silver';"> Sección de contenidos... </div> </pre> <p>En este caso THIS (este mismo elemento) hace referencia al propio div con id="contenidos", que es quien ejecuta los eventos onmouseover y onmouseout. Quiere decir que al propio div se le cambiara el color de borde cuando pase por encima el ratón y cuando deje de pasar por encima.</p>

JAVASCRIPT - FORMULARIOS

ATRIBUTOS

NAME	Nombre del elemento. (Read Only)
ID	Identificador del elemento.
TYPE	Indica el tipo de elemento.
VALUE	Indica el valor que tiene el elemento.

TIPOS - <INPUT>

TEXT	Tipo texto.
NUMBER	Tipo número.
EMAIL	Tipo email. Verifica que el formato esté bien escrito.
PASSWORD	Tipo contraseña. Enmascara el texto introducido con símbolos.
RADIO	Tipo "selecciona una opción".
CHECKBOX	Tipo "marca la casilla".
URL	Tipo URL.
HIDDEN	Esconde el campo, pero sigue estando en el formulario.
FILE → ACCEPT=".jpg, .png, ..."	Tipo archivo. Verifica que el formato esté bien escrito.
BUTTON	Tipo botón. Puedes darle una función con el evento onclick.
RESET	Limpia el formulario.
SUBMIT	Envía el formulario.

RESTRICCIONES

REQUIRED	Indica que el elemento no debe estar vacío.
MIN / MAX	Indica el número mínimo y máximo que se puede introducir.
PATTERN	Indica el patrón que se debe seguir para validar el campo.
MINLENGTH / MAXLENGTH	Indica la longitud mínima y máxima del texto introducido.
CHECKVALIDITY()	Valida si el valor introducido cumple con las restricciones.
SETCUSTOMVALIDITY("mensaje")	Mensaje personalizado que se mostrará en el elemento a validar.
let validación = elemento.VALIDTY	Retorna el estado de validación del elemento.
validación.valueMissing	Retorna TRUE si no hay valor. (REQUIRED)
validación.patternMismatch	Retorna TRUE si el valor no coincide con el patrón.
validación.tooLong / tooShort	Retorna TRUE si el valor es más largo/corto que máximo/mínimo especificado. (MAXLENGTH / MINLENGTH)
validación.rangeOverflow / rangeUnderFlow	Retorna TRUE si el valor es mayor/menor al máximo/mínimo especificado. (MAX/MIN)
validación.typeMismatch	Retorna TRUE si el valor no coincide con la sintaxis del tipo especificado. (TYPE = "EMAIL/URL")

OBTENER VALORES

TEXTO / TEXTAREA	let valor = document.getElementById("texto").value;
RADIOBUTTON	let elementos = document.getElementsByName("name"); for (var i=0; i < elementos.length; i++) { let valor = elementos[i].value; let seleccionado = elementos[i].checked; }
CHECKBOX	let valor = document.getElementById("id").value; let seleccionado = document.getElementById("id").checked;
SELECT	let lista = document.getElementById("idSelect"); let indiceSeleccionado = lista.selectedIndex; let opcionSeleccionada = lista.options[indiceSeleccionado]; let textoSeleccionado = opcionSeleccionada.text; let valorSeleccionado = opcionSeleccionada.value;

Validaciones personalizadas:

<form **ONSUBMIT="RETURN VALIDACION()"**> Se utiliza una función para validar todos los campos.

Si se cumplen las siguientes validaciones se **retornará FALSE**:

	let valor = document.getElementById("campo").value;
valor.length == 0 (texto)	Obliga a que el texto introducido no esté vacío.
isNaN(valor) (número)	Obliga al usuario a introducir un valor numérico.
valor.selectedIndex == 0 (select)	Obliga a seleccionar una opción de una lista.
	let elemento = document.getElementById("campo");
elemento.checked	Verifica si el checkbox ha sido seleccionado.

EXPRESIONES REGULARES

Se validan mediante el método "test()" → **expresionRegular.test(valor)**

/^w+([.-+]?w+)*@w+([.-]?w+)*(\.w{2,10})+\$/	Valida un email.
/^[a-zA-ZáéíóúÁÉÍÓÚüÜñÑ'-.]+s?[a-zA-ZáéíóúÁÉÍÓÚüÜñÑ'-.]*\$/	Valida un nombre.
/^(d{8}[A-HJ-NP-TV-Z] [XYZ]d{7}[A-Z])\$/	Valida tanto DNI como NIE.
/^(0[1-9]) (1[2]d 3[01])(V)(0[1-9] 1[012])2\d{4}\$/	Valida fecha.
/^(+d{1,3}s?)?(d{3,4}[-\s]?){2}d{3,4}\$/	Valida teléfono.

JAVASCRIPT – TRY / CATCH

Try {

Aquí intenta ejecutar el código.

ej: conectar a una base de datos / abrir un archivo / ...

} Catch (error) {

Aquí intercepta el error.

Se puede retornar el mensaje de error que se quiere mostrar o ejecutar otro código.

} Finally {

Aquí siempre se ejecutará un código al finalizar.

}

PHP	
echo / print	Para imprimir datos por pantalla.
Incluir un archivo externo a la web:	
<code><?php include("archivo.html");</code>	Produce un Warning si no encuentra el archivo.
<code><?php require("archivo.html");</code>	Produce un Error si no encuentra el archivo.
FORMULARIOS <form>	
ACTION	Acceder al archivo .php.
METHOD	Método de envío de datos.
FORMULARIOS - METHOD	
GET	Envía los datos mostrándolos en la URL. http://localhost/tutorial/get.php?name=Mac&age=21
POST	Envía los datos SIN mostrarlos en la URL.
RECOGER DATOS (se utiliza el atributo NAME)	
<code><INPUT TYPE="SUBMIT" NAME="enviar" VALUE="procesar"></code>	
<code>\$nombre = \$_GET["enviar"];</code>	Si el form es METHOD="GET".
<code>\$nombre = \$_POST["enviar"];</code>	Si el form es METHOD="POST".
CAMPOS CHECKBOX <code><INPUT TYPE="checkbox" NAME="extras[]" VALUE="garaje" CHECKED>Garaje</code> <code><INPUT TYPE="checkbox" NAME="extras[]" VALUE="piscina">Piscina</code> <code><INPUT TYPE="checkbox" NAME="extras[]" VALUE="jardin">Jardín</code>	
<pre>\$extras = \$_POST['extras']; foreach(\$extras as \$extra) { print("\$extra
\n"); }</pre>	
SESIONES	
<code>session_start();</code>	Inicializa una sesión.
<code>session_register(variable);</code> <code>\$ _SESSION['nombre'] = valor;</code>	Registra / almacena una variable en la sesión.
<code>session_unregister(variable);</code> <code>unset(\$ _SESSION['nombre']);</code>	Elimina una variable de la sesión.
<code>session_is_registered(variable);</code> <code>if(isset(\$ _SESSION['nombre']))</code>	Comprueba si una variable está registrada.
<code>session_destroy();</code>	Cierra una sesión.
COOKIES	
<code>\$cookie_name = "remember me";</code>	Nombre de la cookie.
<code>\$cookie_value = 1;</code>	Valor de la cookie.
<code>\$cookie_expire_time = time() + (24*3600);</code>	Tiempo de expiración de la cookie. (ej: 1 día)
<code>setcookie(\$cookie_name, \$cookie_value, \$cookie_expire_time);</code>	Crear la cookie.

FETCH (Javascript: async - await)	
async function serverFetch(){}	Se crea la función de tipo async.
const respuesta = await fetch(url) ;	Se hace la petición al servidor.
If (!result.ok) throw {ok:false, msg:"mensaje de error"};	Pregunta si ha salido algo mal, si es así lanza un error.
let data = await respuesta.json() ;	Convierte el resultado en json.
FETCH (PHP)	
header("Content-Type: application/json") ;	Dice que este archivo devuelve un archivo de tipo JSON.
echo json_encode(\$variable) ;	Variable que retorna al fetch en formato JSON. La \$variable puede ser: array, string, int...