|  |  |
| --- | --- |
| JAVASCRIPT | PHP |
| Inicio | |
| (En HTML) => <script> -código- </script> | Tanto en HTML como en PHP => **<?php** -código- **?>** |
| Acceso | |
| <script src=”ruta/archivo.js”></script> | <form **action=”ruta/archivo.php”**></form> |
| Declaración de variables | |
| var variable = “”;  let variable = “”; | **$**variable = “”; |
| Arrays | |
| let nombre\_array = [val1, val2, val3, ….];  let primerValor = nombre\_array[0]; // = “val1”; | $nombre\_array **= array(**val1, val2, val3, ….**);**  $primerValor = $nombre\_array**[0];** // = “val1”; |
| Bucles | |
| for (let index = 0; index < valorMáximo; index++){ -código- }  for (let index = valorMáximo; index > 0; index--){ -código- }  array.foreach( (elemento) => { -código- });  while (condición) { -código- } | for ($index = 0; $index < valorMáximo; $index++){ -código- }  for ($index = valorMáximo; $index > 0; $index --){ -código- }  **foreach** ($array **as** $valor) { -código- }  = |
| Métodos para strings | |
| let cadena = “\*\*\*\*\*\*\*”;  let longitud = cadena.length;  let aMinúsculas = cadena.toLowerCase();  let aMayusculas = cadena.toUpperCase();  let letra = cadena.charAt(indice);  let arrayPalabras = cadena.split(separador);  let porcion = cadena.substring(posición inicio, posición final); | **$cadena = “\*\*\*\*\*\*\*”;**  $longitud = **strlen**($cadena);  $aMinusculas = **strtolower**($cadena);  $aMayusculas = **strtoupper**($cadena);  $letra = **$cadena[indice]**;  $arrayPalabras = **explode(separador, $cadena);**  $porcion = **substr($cadena, posición inicio, posición final);** |
| Métodos para arrays | |
| let array = [val1, val2, val3];  let longitud = array.length;  let creaCadena = array.join(separador);  let eliminaUltimoElemento = array.pop();  let eliminaPrimerElemento = array.shift();  array.push(nuevoElemento); // Añade elemento al final  array.unshift(nuevoElemento); // Añade elemento al principio  array.reverse(); // Coloca los elementos al inverso | **$array = [val1, val2, val3];**  $longitud = **count**($array**);**  $creaCadena = **implode(separador,** $array**);**  $eliminaUltimoElemento = **array\_pop(**$array**);**  $eliminaPrimerElemento = **array\_shift(**$array**);**  **array\_push(**$array**, elemento);**  **array\_unshift(**$array**, elemento);**  $arrayInvertido = **array\_reverse(**$array**)**; |
| Funciones | |
| function miFuncion( parametro1, parametro2, … ) { -codigo- } | |

|  |  |  |
| --- | --- | --- |
| 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 | | |
| let enlace = document.getElementById("enlace");  alert(enlace.href); *// muestra http://www...com*  <a id="enlace" href="http://www...com">Enlace</a> | | let imagen = document.getElementById("imagen");  alert(**imagen.style.margin**);  <img id="imagen" **style="margin:0;** border:0;" src="logo.png" /> |
|  | | |
| window.location *= “http://www.paginanueva.com”;*  window.open*(“pagina.html”,”nombrepagina”,”width=500,height=200”);* | | Cambia la dirección de la página (en la misma ventana).  Abre la dirección en una nueva ventana. |
| CREAR ELEMENTOS | | |
| let parrafo = document.createElement("p"); | Creación de un **nodo de tipo** **Element/Tag** que represente al elemento. | |
| let atributo = document.createAttribute("id"); | Creación de un **nodo de** **tipo Atributo.** | |
| atributo.value = “nombre\_id”; | Se define el contenido del atributo. | |
| let contenido = document.createTextNode("Hola Mundo!"); | Creación de un **nodo de** **tipo Text** que represente el contenido del elemento. | |
| parrafo.appendChild(contenido); | Añadir el nodo Text como nodo hijo del nodo Element. | |
| document.body.appendChild(parrafo); | 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 | | |
| var parrafo = document.getElementById("provisional");  parrafo.parentNode.removeChild(parrafo);  <p id="provisional">...</p> | | 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**

**THIS => Hace referencia al elemento que ha provocado el evento.**

<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>

**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.**

|  |  |  |
| --- | --- | --- |
| 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. |
| VALIDAR ELEMENTO | | |
| Validaciones nativas: | | |
| 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");**  letindiceSeleccionado = lista.**selectedIndex**;  let opcionSeleccionada = lista.**options[indiceSeleccionado]**;  let textoSeleccionado = opcionSeleccionada.**text**;  let valorSeleccionado = opcionSeleccionada.**value**; | |

|  |  |
| --- | --- |
| Validaciones personalizadas: | |
| <form ONSUBMIT=”RETURN VALIDACION()”> | Se utiliza un función para validar todos los campos. |