|  |  |
| --- | --- |
| **array\_merge() Function** | **array\_combine() Function** |
| This function merges the two or more arrays. | This array combine only two arrays. |
| This function merges  the arrays such that all the arrays have keys and values. | This function combine the one array containing keys and another array containing values. |
| The arrays are appended at the end of the first array. | The arrays are combined.  https://www.geeksforgeeks.org/difference-between-array\_merge-and-array\_combine-functions-in-php/ |

**join()** function is built-in function in PHP and is used to join an array of elements which are separated by a string.     print\_r(join("+",$arr)); //alias of implode

implode() function returns a string from the elements of an array.

JAVASCRIPT (https://www.w3schools.com/jsref/default.asp)

let color = "Yellow";

document.write(color);

document.write(typeof color); // Yellow string

const car = {type:"Fiat", model:"500", color:"white"};

document.write(car);

document.write(typeof car); //o/p object object object

LOOPS

For, for (key in object), for (let x of cars)

In simple terms forIN iterates over the keys\_IN the array(index)/object(key), whereas forOF iterates over the VALUES\_OF the array(value).

let x = "Volvo" + 16+16; // Volve1616

let x = 16 +16+ "Volvo"; // 32Volve

**EVENTS**

<button onclick =" myFunction() ">Click me.</button>

Onclick = when u click

Ondblclick= when you double click

Oncontextmenu = when you right click

Onmouseenter = when you hover mouse on it

Onmouseout = when you remove mouse from it

Onkeypress = when you press any key. only work on body or form tag

Onresize = when you resize the window

Onscroll= when you scroll

Onchange = when we change the value for example with select list

**ARRAY**

car.length = property returns the length (size) of an array

car.toString() = converts an array to a string of (comma separated) array values.

car.join(“\*”) = same like tostring but in addition you can specify the separator. By default comma separator.

concat() = method does not change the existing arrays. It always returns a new array.

const fruits = ["Banana", "Orange", "Apple", "Mango"];  
fruits.sort();  
fruits.reverse();

|  |  |
| --- | --- |
| Car.pop() method removes the last element from an array and returns the value that was "popped out" | Car.shift() method removes the first array element and "shifts" all other elements to a lower index. |
| Car.push() method adds a new element to an array (at the end) and returns new array length | Car.unshift() method adds a new element to an array (at the beginning), and "unshifts" older elements. |
|  |  |
|  |  |
|  |  |

**STRINGS METHODS**

let text = "we have Italian citizenship.";

document.write(text.**length** ());

document.write(text.**toUpperCase**());

document.write(text.**toLowerCase**());

document.write(text.**includes**("GHIJ")); //returns true if the found (caseSensitive)

document.write(text.**startsWith**("have"));

document.write(text.**endsWith**("ip."));

document.write(text.**search**("zen")); //returns the index number of searched

document.write(text.**match**(/e/g));

document.write(text.**replace**("have","had"));

text = text.**replaceAll**("cats","Dogs");

text = text.**trim**(); // trim() method removes whitespace from both sides

document.write(text.**charAt**(4)); //returns the char at index 4

document.write(text.**charCodeAt**(4)); //returns the ascii code of char at 4

document.write(text.**split**("have"));

document.write(text.**repeat**(3));

document.write(text.**slice**(3,7));

document.write(**String.fromCharCode**(65)); //returns the char from given ascii

MATH METHODS

**NUMBER** **METHODS**

Number() = convert a string\_number into a number

parseInt() = convert a string\_first\_number into number eg(“55 6”.”55 h”,”89.5”,”76j”)

isFinite() = checks if its number or not

isInteger() = checks if its interger or other….

num.toFixed(3) = it will fix float number 67.367878 become 67.368

**DOM**

|  |  |
| --- | --- |
| Method | Description |
| document.getElementById(*id*) | Find an element by element id |
| document.getElementsByTagName(*name*) | Find elements by tag name |
| document.getElementsByClassName(*name*) | Find elements by class name |

|  |
| --- |
| GET METHODS element= document.getElementsByTagName("h3")[0].attributes; |
| innerHTML = gives content + other tags inside. |
| innerText = only text. |
| getAttribute("class") = gives the value of class abc eg class=”abc” |
| attributes = gives all attributes eg class,id,name |
| attributes[0] = attributes according to index |

|  |
| --- |
| SET METHODS element= document.getElementsByTagName("h3")[0].attributes; |
| innerHTML = |
| innerText = |
| getAttribute("class") = |
| attributes = |
| attributes[0] = |

Difference between querySelector(“.ab”) and getElementByClassName(“ab”)

The **classList** Property is a read-only property. The Classlist in JavaScript property allows you to powerfully manipulate the classes