JavaScript Array

JavaScript Array is a single variable that used to store deferent multiple elements.

There are 2 ways to create array:

Var arr = [];

Var arr= new Array();

**JavaScript Basic Array Methods**:

Array.push()  
Adding elements at the end of array.

**Syntax: arr.push(‘a’,’b’);**

Parameters: Items to be added in the array

Array.unshift()  
Adding the elements at the front of array.

**Syntax: arr.unsift(‘a’,’b’);**

Parameters: Items to be added in the array

Array.pop()  
  
Removing the elements from the end of array.

**Syntax: arr.pop();**

Parameters: It’s take no parameters.

Array.shift()  
  
Removing the elements from the beginning of array.

**Syntax: arr.shift();**

Parameters: It’s take no parameters.

Array.splice()  
  
Insertion and removal in between an array.

**Syntax: arr.splice(start, deletecount, item1, item2, …);**

Parameters:

Start- Location at which to perform to operation.

Deletecount: number of elements to be deleted. If no element to be deleted then pass 0.

Item1, item2…: This is an optional parameter.

These are the elements to be inserted from location start.

JavaScript Array Functions:  
**Array.prototype.every()**

function checkless100\_every(arr){

return arr.every(function(element){

return (element < 100);

});

}

console.log(checkless100\_every([30,60,90])); // true

**Array.prototype.some()**

function checkArrSome(arr){

return arr.some(function(ele){

return (ele > 100);

});

}

console.log(checkArrSome([30,60,20])); // false

**Array.prototype.filter()**

var result = [

{rollno:'100',percentage:'50'},

{rollno:'100',percentage:'40'},

{rollno:'100',percentage:'30'},

{rollno:'100',percentage:'60'}

]

function knowgreatermore60\_filter(arr){

return arr.filter(function(temp){

return Number(temp.percentage) >= 50 ? true : false;

});

}

console.log(knowgreatermore60\_filter(result));

**Array.prototype.map()**

var result = [

{rollno:'100',percentage:'50'},

{rollno:'100',percentage:'40'},

{rollno:'100',percentage:'30'},

{rollno:'100',percentage:'60'}

]

function adddivsionByMap(arr){

return arr.map(function(newarr,i,arr){

newarr.dictaintion = (Number(newarr.percentage) >= 75 ) ? true : false;

return newarr;

});

}

console.log(adddivsionByMap(result));

JavaScript | Date.parse()

Date.parse() is the inbuilt function of JavaScript it helps us to know exact number of milliseconds that have passed since mid night, January 1, 1970, till the date we provide.

**Syntax: Date.parse(datestring)**

Incase datestring is invalid then it return **NaN**

var date = "July 07, 1981 12:30 PM";

var mili = Date.parse(date);

console.log(mili);

output: milliseconds in numbers

JavaScript | date.toLocaleDateString()

date.toLocalDateString() is the inbuilt function of JavaScript which is used to convert a date to a string.

**Syntax: dateObj.toLocaleDateString([Locals],[options])**

var dateobj = new Date();

var options = {weekday:"short",year:"numeric",month:"short",day:"numeric"}

console.log(dateobj); // Tue Apr 16 2019 14:41:53 GMT+0530 (India Standard Time)

console.log(dateobj.toLocaleDateString('en-us',options)); //Tue, Apr 16, 2019

console.log(dateobj.toLocaleDateString('en-us')); //4/16/2019