**JS Flashcard**

**MDN JavaScript Guide**

[**https://developer.mozilla.org/en-US/docs/Web/JavaScript**](https://developer.mozilla.org/en-US/docs/Web/JavaScript)

**JavaScript Fundamental part 1**

1. What is JavaScript?
2. What is the role of js in web development?
3. Where JS can be used?
4. What are JS versions?
5. What is new in ES6? (might become an interview questions)
6. Where we can put js code in html? How to import js file?
7. Is semicolon mandatory in js?
8. Should use double quotes vs single quotes?
9. How to print alert on browser?
10. How to log multiple values in console?

Basic

Values and Variables

MDN variable documentation:

<https://developer.mozilla.org/en-US/docs/Learn/JavaScript/First_steps/Variables>

1. What is a value?
2. What is variable? How to declare a variable? Why do we need it?
3. What are naming conventions for variable in js?

(camelCase, shoud not start with numbers, do not use reserved keywords)

Data types

1. How many types of values in JS? What are they?
2. How many date types?
3. What is dynamic typing? What it achieves?
4. How to comment code of one line? A block?
5. How to check the type of a variable/value? Try print out all data types
6. What is type of null values? And why is it?

3 ways of declaring variables (var, let, const)

1. When let and const were introduced?
2. What is mutate the variable? Which keyword should be used? Which assignments can be applied?
3. When should use const? which case can be used? Which cases cannot be used?
4. Difference between let (block scoped) and var (functional scoped)?
5. How to create a global variable?

Basic Operators

1. Addition, Reduction, multiply, divide, power
2. contact string
3. Assignment operators
4. Comparison operators

Operators

1. Operator precedence?

<https://developer.mozilla.org/enUS/docs/Web/JavaScript/Reference/Operators/Operator_Precedence>

Strings (template literals)

1. How to create template string? Syntax?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Template_literals>

1. How to create multi-line with template string?
2. How to use expression interpolation with template string?

If/else control

1. Syntax for if-else state statement
2. If-else if

Type conversion vs coercion

1. What is type conversion?
2. What is type coercion?
3. Which function can be used to convert string to number? What if the string is not a number? How to convert number to string?
4. What is NaN? What is the type of NaN?

Js can convert values to number/Boolean/string

1. Which operator can be used to convert number as string? Which one cannot be used?

Truthy and Fasly values

1. What are falsy values? How many are they?

(0, ‘’, undefined, null, NaN)

1. What are truthy values?
2. When falsy values will be covercion and being used?

Equality operators “==” vs “===”

1. What is === strict equality operator?
2. == does type coercion for which type?
3. What is prompt() function?
4. What is negation of “===”?

Logical operator (AND, OR and NOT)

1. What are logical operators? || && !

Switch statement

1. What is switch statement syntax?
2. <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/switch>
3. Which case will it maps to if there are two same cases?
4. How the expression and case value compared?
5. Anything else can be used rather than break?
6. How block scope variable can be used in swtich statement?

Expression vs statement:

1. What is an expression?
2. What is a statement?

Conditional Operator(ternary)

1. Syntax?
2. What are possible falsy values?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Conditional_Operator>

JavaScript History

1. First version of JavaScript in 1995? Mocha
2. JavaScript vs Java? 1996 renamed to JavaScript
3. 1997, standardize the language, ECMA (ES1)
4. 2009 ES5
5. 2015 ES6: Biggest update to the language ever
6. 2016 – present annual releases
7. Modern JS still support the old versions (backwards compatible)
8. How to use modern JS?

During development: use the latest google chrome

During production: Use **Babel** to **transpile** and **polyfil** your code (converting the code to ES5 to ensure browser compatibility for all users)

ES5:

1. Fully supported in all browsers
2. Ready to be used today

**JavaScript Fundamental part 2**

Activating strict mode

1. What is strict mode? How to activate it?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Strict_mode>

Functions

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Functions>

1. What is a function?
2. How to define a function? Which parts it has?
3. Parameters in functions are passed by value? What does it mean for primitive and object?
4. What are anonymous functions?
5. What is function expression?
6. What is hoisted? When it can be achieved?
7. What variables cannot be accessed in a function?
8. What variables can be accessed in a function?
9. What if a function expression does not return value? What will be the value?
10. What is the difference between function declaration vs function expressions?

Functions are values in JavaScript

1. What if there is argument passed in a function which returns a number?

Arrow functions

1. When arrow functions were introduced?
2. What is arrow function? What does it have anything to do with function expression?

Function call another function

1. What is DRY principle or don’t repeat yourself principle?

Arrays

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array>

1. How to create an array object?
2. How to access element in the array?
3. How to get access to the length of the array?
4. How to retrieve last element of an array?
5. How to mutate element in the array?

Basic Array operations (methods)

1. What is operation to add one element in the end of array?
2. What is operation to add one element in the beginning of array?
3. What is operation to remove the element in the end of array?
4. What is operation to remove the element in the beginning of array?
5. What is operation to get the index of element/ non-existing element?

Introduction to objects

1. What is object liter syntax?
2. Difference from array?

Dot vs Bracket Notation

1. How to access properties with both notations?
2. What are the difference?
3. How to add new properties in the JSON object?

Object methods

1. How to add a function as a property in an object? Can function declaration be used? Can arrow functions be used?
2. How to use bracket notation to access the function in the object?
3. How to access the property within the same object?

Iteration for loop

1. For loop syntax

Looping arrays, breaking, continue

1. Which property is used often in for looping an array?
2. How to add element in array in for loop?
3. What is break and continue?

Looping backwards and loops in loops

1. Looping backwards
2. Embedded loops

While loop

1. While loop syntax

DEVELOPER TOOLS:

Prettier default format

TODO highlight

Auto refresh page:

1. Node.js npm
2. Live server

**HTML and CSS Crash course**

1. What is the main file name for HTML?
2. HTML stands for ? What is used for?
3. Composed of what?
4. What is the tag that all HTML files start with?
5. What are the two elemens in html? And what are they used for?
6. What is h1, what is p?
7. What are attributes in html?
8. What is anchor element? What is href stands for?
9. What is block element and inline element?
10. Element for image? What its attribute?
11. What are two important attributes for all elements?
12. What one is good to be used for styling?
13. What is creating boxes? Which element to use?
14. How to separate two words in html?
15. What element is used for user to enter values? What attribute is in this element?
16. How to link css file to the html fiile? Which element to use?
17. CSS attributes? Font? Color?
18. What the style for child and its parent?
19. How to specify style using class attribute?
20. How to specify style using id attribute?
21. What is border? Is it inherited?
22. What is global attribute? What is border-box?
23. CSS box model has what attributes?

* Content
* Padding
* Margin
* Fill area
* Border

1. How can selectively add style for margin and padding?
2. How to make text to be in the center?
3. How to comment out css code?
4. Selectors css? What are they?
5. Are input/button inline or block element?
6. What is a child selector?
7. How to set the image size? Why we should do it? What if we only set width or height?

**JavaScript in Browser: DOM and Events fundamentals**

Number guessing project

1. Document.querySelector is for what?

<https://developer.mozilla.org/en-US/docs/Web/API/Document/querySelector>

1. What is css selector?

<https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Selectors>

* How many basic types of css selector?
* What is grouping selector
* What is combinators

1. How to link js with html? How to link css with html?
2. What is DOM?

<https://developer.mozilla.org/en-US/docs/Glossary/DOM>

1. What is tree structure? And how element is represented? Draw one
2. What is the entry point to the DOM? Give an example?
3. What is difference between DOM vs JS?

JavaScript

1. Add click events how? Which function can be used for that action?

<https://developer.mozilla.org/en-US/docs/Web/API/EventTarget/addEventListener>

1. Which attribute can we use for getting the value?
2. How to generate a random number? How to make it an integer?

**How to manipulate css attribute?**

1. Which attribute to use to get the css property?
2. How to change a specific attribute for an element?
3. When changing the attribute value, what is the type of the value? Must be?
4. What is the difference between value vs textContent?
5. What is DRY principle? Why should we use this principle?
6. How to erable one class temparaliy? Which method to use? Which attribute to use?
7. How to add one class? Which method and attribute to use?
8. In addEventListener method, which function expression to use? Difference between functionName() vs functionName?
9. How to add an event gloabally?
10. How to check whether the element has a certain attribute?
11. How to add an event for pressing a key?
12. How to get access to the key information when pressing it? Where the information is stored?

* An object (keyboardEvent) will be created for the event, and it can be passed in to the function expression as an argument
* Note: this definition can only happen when

**Pig game project**

1. How to get the element by id? Which method to use on document? What is the difference between querySelector?

<https://developer.mozilla.org/en-US/docs/Web/API/Document/getElementById>

1. How to change attribute in an element? Which method to use?
2. How to use combination css selectors?
3. Toggle method?

<https://developer.mozilla.org/en-US/docs/Web/API/Element/classList>

**JS Behind the Scenes**

What is JS?

1. High Level
2. Garbage collection
3. Interpreted or just in time compiled
4. Multi-paradigm

* An approach and mindset of structuring code, which will direct your coding style and technique

1. Procedural programming
2. Object-oriented programming
3. Functional programming
4. Prototype based object oriented
5. First-class functions

Functions are simply treated as variables

1. Dynamic typed
2. Single-threaded
3. Non-blocking event loop

JS Engine and Runtime

1. Browser has JS engines
2. What are components in a JS engine? (Call stack, Heap)
3. Compilation vs Interpretation vs Just-in-time Compilation
4. Is JS an interpreted language?
5. Modern Just-In-Time compilation of JavaScript? Engine

* Parsing (Where it happens?AST)
* Compilation (Where it happens? Just-in-time)
* Execution (Where it happens?)
* Optimization (What is this step? Why is it important? Where it happens?)

1. JS runtime?
2. What does JS Runtime includes? (JS engine, Web APIs, Callback queue)
3. Event loop why important?

Execution in JS engine

1. What is execution context?
2. What are there in an execution context?

* Variable Environment

Let,const, var declarations

Runctions

Argument object

* Scope chain
* This
* What is special for arrow functions?

1. Execution steps?

Right after the compilation

1. Creation of global execution context (for top-level code): exactly one, default context
2. Execution of top-level code
3. Execution of functions and waiting for callbacks (One execution context will be created for each function)

* What is callback function?

<https://developer.mozilla.org/en-US/docs/Glossary/Callback_function>

1. What is call stack?

Place where execution contexts get stacked on top of each other, to keep track of where we are in the execution

Scope and Scope in js (和JAVA的scope很相似)

1. Lexical scoping:

Scoping is controlled by placement of functions and blocks in the code

1. Scope:

Space or environment in which a certain variable can be declared. There is global scope, function scope and block scope (es6)

* A function can be defined in a if block

1. How many types of scope?
2. Which type of variable are block scoped?
3. When a function is block scoped?
4. What is scope chain? What is variable look up in scope chain? What aboue the other direction?
5. Difference between const, let and var?

* Var ends up in the closest function scope

1. What is scope chain vs call stack?
2. What if two variables have same name but in different scope in the scope chain?

Hoisting and the TDZ

1. What is JS hoisting?

<https://developer.mozilla.org/en-US/docs/Glossary/Hoisting>

<https://www.w3schools.com/js/js_hoisting.asp>

1. Is declarartion hoisted?
2. Is initialization hoisted?
3. What is let and const hoisting?
4. What is the difference between function declaration, function expressions, arrow functions?

* When assigning the function to const/let: Reference error
* When assigning the function to var: TypeError

1. Best practices with hoisting:
2. Always declare varibales/functions at the top of the block
3. Don’t use var to declare variables
4. Global variable window

Var varible will be created as a property on the window object

This keyword

1. What is this keyword?

<https://www.w3schools.com/js/js_this.asp>

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/this>

* Special variable created for every execution context (every function)
* Not static. Depends on how the function is called

1. This is NOT static

* Depends on how the functions is called

1. Return this in a function?

* Under strict mode
* Not under strict mode
* Arrow function? Do not have their own keyword, it uses its parent function or parent scope

This = <this of surrounding function (lexical this) >

1. Event listener this?

* This = <DOM element that the handler is attached to>

1. Understand this keyword
2. <https://www.codementor.io/@dariogarciamoya/understanding--this--in-javascript-du1084lyn>
3. <https://www.codementor.io/@dariogarciamoya/understanding-this-in-javascript-with-arrow-functions-gcpjwfyuc>

Regular functions vs arrow functions

1. Arrow function does not have its own this keyword

* The arrow function inside an object, the this keyword does not refer to the object, since {} are used to define an object literal
* Var variable in the gloabal scope that is considered that it creates a property on the global variable (e.g. window) (we should not use var to declare variables)

1. When a function is declared as the property of an object, then this refers to the object. What if there is a function inside that function, then how to get access to this? How to get access to the object?

* use a variable assigned with this and use the variabvle in the function
* use arrow function

1. Argument keyword

Primitive vs object

1. Primitive types:

Where are they stored?

* Number
* String
* Boolean
* Undefined
* Null
* Symbol
* BigInt

1. Object (reference types)

Where are they stored?

* Object
* Array
* Function

1. How to copy objects?

* Ojbect.assign(); this function only creates a shallow copy
* What is difference between shallow copy vs deep copy

**JS Data structure**

Destructuring assignment

The **destructuring assignment** syntax is a JavaScript expression that makes it possible to unpack values from arrays, or properties from objects, into distinct variables.

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Destructuring_assignment>

Destructuring arrays

1. How to assign values from array to variables?
2. How to swap values? Easy way?
3. How to return multiple values from a function?
4. How to escape an element?
5. How to assign values when dealing with a nested array?
6. What are default values? When to use them?

Destructuring objects

1. How to get properties in the object? Syntax?
2. How to rename properties when assigned? How to set default values?
3. Mutate varibales using destructuring? Reassign values from properites in an object?
4. How to assign values which is nested object?
5. Can be passed in function as a parameter? Does the order of properties matter?

Spread operator

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Spread_syntax>

definition: work with iterable

1. How to use it to pass all elements of array?
2. How to make a shallow copy of an array?
3. How to join two arrays?
4. How to use it separate elements in Iterables?

* Spread operator will separate elements in arrays, strings, maps, sets NOT objects (However starting from 2018, objects also works with the spread operator)
* This operator cannot be used in embedded expression (template literals)

1. How to pass in array as parameter list where the function takes a list of variables?
2. How use it to copy an object?

Use spread operator pack elements (Rest elements)

1. When to use it as packing
2. When to use it as upacking

* Rest must be the last element

1. How it works with objects?
2. How it works as function parameter?

Short Circuiting

Definition <https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Logical_AND>

1. What is result of 3 || “Jonas”?
2. What is result of 0 && “Jonas”? 7 && “Jonas”
3. What is result of (undefined || null)?
4. What are falsy values?

* Undefined, null, “”, 0, NaN

1. OR will return fist trusy value or the last value
2. && WILL return first falsy value or the last value

Nullish Coalescing Operator

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Nullish_coalescing_operator>

1. ES2020 ?? check whether it’s these two values?
2. Nullish meaning null or undefined.

For of looping

可以和Java中的enhanced loop进行比较

1. What is the syntax of for of loop?
2. How to get the index of each item?

Enhanced object literals

1. Reference a property in an object?
2. How to define a function as a property in an object?
3. How to reference a property name?

Optional chain

可以和Java8 中的Optional class进行比较

文档

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/Optional_chaining>

1. Why does this optional chain exist? When it’s introduced?
2. What is the syntax?
3. How is it can be used on methods?
4. How is it can be used on arrays?

Looping Objects

1. How to retrieve the keys of an object?
2. How to retrieve the values of an object?
3. How to retrieve the entries of an object?
4. How to retrieve a certain property of an object?
5. What is the difference between object.entries() and array.entries()?

Sets

1. How create a new set object? What can be put to initiate the set?
2. What if a string to be used to create a set?
3. How to check the number of elements in set?
4. Methods on set? Add, remove, check a value exist, remove all?
5. How to convert a set to an array? Array to a set? Iterable
6. When should use set vs array?

Maps

1. Syntax of creating a map object
2. How to add new key value pairs? Using which method? Can the method be chained?
3. What values can be keys?
4. Which method to retrieve value?
5. Which method to check a value? Check the key or value?
6. Which method to delete value? What is the parameter to be passed in?
7. How to check number of elements?
8. How to remove all elements?
9. Can array be the key in the map? How to make sure to retrieve the mapped value for an array?
10. Array convert to map? Mapp conver to array? Convert keys into an array? Convert values into an array?
11. How to convert object to map?

What data structure to use? (object, array, map, set)

1. Sources of data?
2. Data structure
3. Simple list? Array, set (when no need to describe properties)

**Array**: keep order; manipulate data

**Set**: unique values only; high-performance; remove duplicates

1. Key/value pairs? Object, map

Object:

more traditional key/value pairs; easier to write and access values with . []

when: include functions; working with JSON

Maps:

better performance; keys can have any data types; easy to iterate; compute size

when: map key to values; needs keys that are not strings

1. How to use map vs object to record times of appearance for one property?

**Strings**

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String>

1. How to get access to one of the character in a string? String variable and string literals
2. How to check the string length?
3. How to check a certain character’s index (first time appears)?
4. How to search a word in a string?
5. How to search the last character appeared the index?
6. How to extracts a section of a string and returns it as a new string? endIndex counts or not?
7. What if the index passed in slice method is a negative number? What is the difference when passing one index from two indeces?
8. How to create a string object? What is the difference between string literal and string object?
9. How to switch all letters to lower case or upper case?
10. How to make sure a word’first letter is capitalized and the rest are lowercase?
11. How to compare email addresses? Which methods can be used?
12. How to replace characters in string? What if the words to be replace appear multiple times? Which function to use then? How to use regular expression here?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_Expressions>

1. How to check whether the string contains a substring? Methods can be used?

* Practice: airplane baggage check in

1. How to split strings into pieces? What is the return result?
2. How to combine strings into one? String arrays?

* Caplitalize names in a function

1. Padding strings?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/String/padStart>

How to make credit card number only show last 4 digits

1. How to repeat characters?

**Functions**

Default parameters

1. When it will be used? How?
2. When there are multi default parameters
3. How to skip a parameter? What if the third parameter has dependency on the second paaramter? Whil second parameter is not set?

How passing arguments works: value vs reference

1. What is the difference between JAVA when it comes to this topic?
2. Is primitive parameter passed by value?
3. Is reference parameter passed by value?

First-Class and High-Order Functions

1. First-class functions?

* Functions are simply values
* Another type of object

Functions can be stored in variables

Functions can be passed as parameters

Functions can be returned from another function

1. Higher order functions?

* Receives/returns a function
* Callback function? Returned function

Functions accepting Callback functions

1. How to receive a function as parameter? How to print the funciton’s name?
2. What is callback function?

* A callback function is a function passed into another function as an argument, which is then invoked inside the outer function to complete some kind of routine or action.

Functions returning functions

1. How to create a function returning a function
2. Why is it important? Functional programming?
3. Use arrow functions to implement the functions returning functions

Call and apply methods

1. How to add a new property in an object?
2. How the call function to be used?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Function/call>

1. How the apply function to be used?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Function/apply>

1. What apply and call difference?

Bind method

1. What is bind method?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Function/bind>

1. How to predefine the parameters in the bind method? Does the order matters?
2. How to work with eventlistener?

* Note: when the function is called, we need to make sure which this keyword refers to
* Note: if a function is created in an object, but it cannot guarantee the function key word is the object when the function is invoked

1. What is partial application?

* Note: this is preset the parameter
* Implement function returning function

IIFE

1. What is IIFE?

<https://developer.mozilla.org/en-US/docs/Glossary/IIFE>

1. What is the design pattern for IIFE?
2. What is the similarity with JAVA?
3. What is the syntax? Two parts?
4. Can vairables defined in the function be accessed from outside? What is the mechanism achieved here?
5. Any other variables cannot be accessed from outside?

Closures

1. What are closures?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Closures>

A closure is the combination of a function and the lexical environment within which that function was declared.

1. What is Lexical scoping?
2. How the execution context work for closures?

* Note: a function always has the access to the variable environment of the execution context in which it was declared
* Closure: VE attached to the function.
* Closusre is created automatically by js engine

1. How to check closure for a function?

* Console.dir(func);

1. What if a function variable is re-assigned declaration?
2. Does closure has priority in scope chain?

**Working with Arrays**

Simple array functions

1. Array methods documentation

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array>

1. Slice() what does it do? Aruguemets passed? Negative number? To copy an array?

* **slice does** not alter the original array. It returns a shallow copy of elements from the original array. Elements of the original array are copied into the returned array as follows: For object **slice** copies object references into the new array.

1. Splice()? How to use it? When only pass one number argument?

* Does it mutate array?

1. How to reverse an array?

* Does it mutate array?

1. How to concat?

* Does it mutate array?

1. What does join() do?

Arrays for each

1. forEach method in arrays

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/forEach>

1. Syntax? Return value? Parameters?

* The order of parameters are important

1. What is the difference between forEach and for loop?

* We cannot break forEach loop

forEach with Maps and Sets

1. What are the parameters in forEach method on a map?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Map/forEach>

1. Are the parameters mandatory?
2. What are the parameters in forEach method on a set?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Set/forEach>

1. Why there are still three parameters for set forEach? What is the difference between the first two?

Create DOM

1. insertAdjecentHTML <https://developer.mozilla.org/en-US/docs/Web/API/Element/insertAdjacentHTML>
2. Different positions for the function insertAdjecentHTML: afterbegin, beforeend

Data transformations: map, filter, reduce

Parameters, return values?

1. Map method on array

Parameters, arrow functions

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/map>

1. Filter method on array

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/filter>

1. Reduce method on array

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/reduce>

find the max value in an array

Method chaining js

1. For map, reduce, filter can do pipeline like stream in java
2. What is pipeline?
3. How to debug the results of one of the method in the chain?
4. How to use method chaining?

* Optimize otherwise it may cause performance issue
* Do not call the method that mutate the original array

Find method for array

1. Find syntax

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/find>

1. Parameters, return values?

* Can be used to check an object in an array

Implement login

1. Opacity css ?
2. Element blur() function remove mouse focus
3. findIndex method on array

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/findIndex>

Array some and every

1. some

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/some>

1. every

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/every>

1. Separate callback function

Used to be passed as an argument in the functions

Flat and flatmap

1. Flat

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/flat>

1. flatMap

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/flatMap>

Arrays sorting

1. sort()?

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/sort>

1. Does it mutate the original array?
2. How to control the ascending/descending order?

Creating and filling arrays

1. Directly? Using new?
2. Fill method

<https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/fill>

* Does it mutate the original array?

1. How to make a copy of an array?

* Arr.slice()
* **Array.from()**
* **[…array]**

1. How to use Array.from() function?

* What is the callback function in Array.from()?

Array methods

1. Mutate original array:

Push, unshift, pop, shift, splice, reverse, sort, fill

1. A new array

Map, filter, slice, concat, flat, flatMap

1. An array index:

indexOf, findIndex

1. An array element:

Find

1. Know if array includes:

Includes, some, every

1. A new string:

Join

1. Transfor to value:

Reduce

1. Loop over: forEach

**Numbers, Dates**

Converting and checking numbers

1. Numbers are decimals
2. 0.1 + 0.2 === 0.3 ???
3. How to convert strings to numbers?
4. How to parse strings to numbers

String contains number

* To integer
* To floating

1. How to check whether it’s a number NaN?
2. How to check whether it’s a numbeR?
3. How to check whether it’s a number integer?

* 23.0