1. How is “==” different from “===”?
2. What is a difference between function’s parameter and function’s argument?
3. How do you define anonymous function?
4. What is for-in loop and for-of loop?

For-in loop is used to read the index number of the elements in the array.

For-of loop is used to read the elements of array.

1. What are map() and forEach() methods?

These are array methods. map() method receives function as a parameter and returns a whole new array executing the provided function.

forEach() method returns an undefined value.

map(currentElement, IndexOfElement, Array)

1. Reduce method default parameters

reduce(accumulator, currentElement, currentIndex, Array) //call back function

Is useful to get the sum, average, mean, deviation of the elements of a given array.

reduce() method as call back function has an initial value at the last of the expression.

Ex: reduce(accum, currEle, Index, arr), initial value)

1. ECMAscript6 major changes…
   1. “let” and “const”

Variable declared using var and let keyword can be changed and modified whereas const is more like a constant which if used, does not allow user to alter the values assigned passing through it.

Also, var has function scope which means that the variable declared using it inside of a function or any loop

But let and const have block scope which means that if a variable is declared using ‘let’ and ‘const’ outside a block ‘{ }’, it cannot be passed outside the block for use.

* 1. Template literals

`${}`

Makes code easily readable. Easy to use and implement if comfortable using it. Use of backticks, inside where a variable is referred by using a dollar($) sign outside the curly braces and called inside those braces.

* 1. Default parameters/arguments

Values can be assigned to the parameters by default while creating a function to make things easier. It has large scope. In a project if I’m taking more than two parameters in a function and I want one or two of those parameters to have certain values throughout the function expression, I can assign default value to those parameters to make things easier and lighter for me.

* 1. Rest operators
  2. Destructuring
  3. Object properties
  4. Arrow function

Use of fat arrow symbol(“=>”) to refer to a function expression make things easier. But key difference between fat arrow function and traditional function is that we cannot call the function prior to the function expression while using the fat arrow expression. While in a traditional function I can call the function to console before or after the function expression, that would not matter. However, in most cases we call the function after the end of the expression to ease the debugging process and prevent future possible error.

But with arrow function we MUST call the function after the function expression. So, primarily the main advantage of using the array function is to improve the readability of the function and save time as it shortens the code.

Note: Cannot use “this.” keyword as argument

#traditional

var a,b;

function sum(a,b){

    return result=(a+b);

}

console.log(sum(2,3));

#fat-arrow function

const sum = (a,b) => `result=${a+b}`;

console.log(sum(3,2));

* 1. Spread operators