A FÜGGVÉNYEK

/\*

write a function that has 3 arguments and returns the average of its arguments

Sample input: 1, 2, 6

output: 3

\*/

console.log('Első feladat');

function avg3(a,b,c){

return (a+b+c)/3;

}

console.log(avg3(5,6,7));/\*

write a function that has an undefinite number arguments and returns the average of its arguments

Sample input: 2, 4, 6, 8

output: 6

input: 1, 2, 6

output: 3

\*/

console.log('Második feladat');

function avg(...args) {//átadott paraméter tömb, benne van a függvény összes paramétere

var sum =0;

for(var i=0; i<args.length; i++){

sum+=args[i];

}

return sum/args.length;

}

console.log(avg(2, 4, 6, 8));

console.log(avg(1, 2, 6));

/\*

Write a function that calculates the sum of it's first 2 arguments and the average of the rest of it' arguments

sample input: 1, 2, 3, 4, 5

output: 7 (1 + 2 + (3 + 4 + 5 ) / 3)

\*/

console.log('Harmadik feladat');

function sumAvg(x,y,...others) {

var sum2 =0;

for(var i=0; i<others.length; i++){

sum2+=others[i];

}

return sum2/others.length + x + y;

}

console.log(sumAvg(1, 2, 3, 4, 5));

/\*

Write a function that returns a function that returns it's argument multiplied by 10

sample input: -

output: a function

\*/

console.log('Negyedik feladat');

function fMultiply10() {

return function(x){

return x \* 10;

}

}

/\* arrow-ot használva:

return x=>10\*x;

\*/

let func = fMultiply10();

console.log(func(10));

//console.log(fMultiply()(10));

/\*

write a function that accepts an x argument and returns a function that returns it's argument multiplied by x

sample input: 10

output: a function

\*/

console.log('Ötödik feladat');

function fMultiplyX(x) {

return function(y){

return x \* y;

}

}

console.log(fMultiplyX(10)(20));

a TÖMBÖK

console.log('Gyakorlás');

console.log('push');

let array = [1,2,3];

array.push(10);

console.log(array);

let p=array.pop();

console.log(p, array);

let s=array.shift();

console.log(s, array);

let us = array.unshift();

console.log(us,array);

array.forEach((x, idx)=>console.log(`array[${idx}]=${x}`));

let array2 = array.map((x,idx) => x\*10);

console.log(array2);

console.log(array);

let s2 = array.reduce((sum, element)=>sum + element, 0);

console.log(s2);

let szorzat = array.reduce((sz, elem)=>sz\*elem,1);

console.log(szorzat);

/\*

With the help of the reduce method find the minimum of the array's elements

JS reference:

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\_Objects/Array/Reduce

\*/

array=[-22, 56, 78, -3, -5, 12, 7, 8, -365];

let min = array.reduce((min, element)=> min > element ? element : min, array[0]);

console.log(min);

/\*

with the help of the reduce method reverse an array

JS reference:

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\_Objects/Array/Reduce

\*/

let reverse = array.reduce((reversed, element)=>{

reversed.unshift(element);

return reversed;

},[]);

console.log(reverse);

/\*

with the help of the map method reverse an array

https://developer.mozilla.org/hu/docs/Web/JavaScript/Reference/Global\_Objects/Array/map

\*/

let reverse2 = array.map((elem, index)=> array[array.length - index-1]);

console.log(reverse2);

/\*

write a function that returns a deep copy of an array that contains nested arrays

You can use the Array.isArray(value) method to decide if value is an array

JS reference:

https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\_Objects/Array/isArray

\*/

function deepCopy(array) {

return array;

}

let arrayToCopy = [1, 3, 5, [3, 3, 3],

[2, [11, 12]], 24

];

let copy = deepCopy(arrayToCopy);

arrayToCopy[4][1][1] = 1000;

console.log(JSON.stringify(arrayToCopy));

console.log(JSON.stringify(copy));

//the copy should not change!!!

/\*

write a function that returns a deep copy of an array that contains nested arrays

without using loops

\*/

function deepCopy2(array) {

return array;

}

arrayToCopy = [1, 3, 5, [3, 3, 3],

[2, [11, 12]], 24

];

copy = deepCopy2(arrayToCopy);

arrayToCopy[4][1][1] = 1000;

console.log(JSON.stringify(arrayToCopy));

console.log(JSON.stringify(copy));

//the copy should not change!!!

Házi feladat