Room espace oyunu

Yumurta ve çiçeği kırınca cevabı bilmeden ekemez.

## document.write() sadece gösterir

document.getElementById("demo").innerHTML = 5 +6 içeriği değiştirebilir

alert(5 + 6); açılan pencere

var x, y, z;    // Statement 1  
x = 5;          // Statement 2  
y = 6;          // Statement 3  
z = x + y;      // Statement 4

<button onclick="document.getElementById('demo').innerHTML = Date()">The time is?</button>

JAVA NOTES: 1. // [comment] Single line comment.

2. /\* [comment] \*/ Multi line comment.

3. public This can be imported publically. 4. import [object].\* Imports everything in object. 5. static Going to be shared by every [object]. 6. final Cannot be changed; common to be defined with all uppercase. 7. double Integer with numbers that can have decimals. 8. ; Put after every command. 9. String Just a string of characters. 10. Private Can only be changed by a method. 11. int Can store numbers from 2^-31 to 2^31. 12. fields are attributes 13. boolean Can have true or false as the value. 14. { } These are used to start and end a function, class, etc. 15. byte These can store from -127 - 128. 16. long Can store numbers from 2^127 to 2^-127. 17. char Just lets you put in one chracter. 18. double 64-bit number with decimals. 19. float 32-bit number with decimals. 20. protected Can only be accessed by other code in the package. 21. Scanner This lets you get user input. 22. new [object constructor] This will let you create a new object. 23. [System.in](https://www.youtube.com/redirect?q=http%3A%2F%2Fsystem.in%2F&stzid=UgyFPVj3fS5kKsWxkId4AaABAg&redir_token=QUFFLUhqbEtreEt1OVItSkVrUWFpb1ZYZmJYbXBFMTBlUXxBQ3Jtc0ttVUJUWlZoaXlCWTJiLUpMZnEzN2J5NzJ2eGdvQ2xlM2xTR25Ca1poWkNNMXhhOUptSDZZWmlsNGVNVnVSMnJueEVNU05hZWRzRERWRnh6bmdWV2FHLUFGdHdKR3N1YXg5dFpfRWNteS0zSmxTeUpqQQ%3D%3D&event=comments) This lets you get data from the keyboard. 24. public [class]() This will be the constructor, you use it to create new objects. 25. super() This will create the superclass (the class it's inheriting). 26. extends [class] Makes the object a subclass of [object], [object] must be a superclass. 27. ++ Will increment the amount. 28. -- Will decrement the amount. 29. += [amount] Increment by [amount] 30. -= [amount] Decrement by [amount] 31. \*= [amount] Multiply by [amount] 32. /= [amount] Divide by [amount] 33. System.out.println([text]) Will print something to the output console. 34. + Can be used for concatenation. (ex. "6" + [var\_here]) 35. public static void main(String[] args) This is your main function and your project will start in here. 36. System.out.print([text]) This prints stuff but there is no line break. (/n) 37. \n Called a line break; will print a new line. 38. \t This will print a tab. 39. if ([condition]) This will make it so if [condition] is true then it'll keep going. 40. && This means and. 41. ! This means not. 42. || This means or. 43. == This means equal to. 44. < This means less than. 45. > This means greater than. 46. >= This means greater than or equal to. 47. [inputVarHere].hasNextLine() This will return if there is a next line in the input. 48. this Refer to the class that you are in. 49. [caller].next[datatype]() This will get the [datatype] that you somehow inputted. 50. Create getters and setters This will create the get methods and set methods for every checked variable. 51. [caller].hasNext[datatype]() This will return if it has the correct datatype within the input. 52. overloading If you have different parameters you can call them whatever way you want. 53. parameters These are the inputs of your function. 54. ([datatype])[variable] This will convert [variable] into [datatype]. Also known as casting. 55. Math.random() Generate an extremely percise string of numbers between 0 and 1. 56. Primitives Just the basic data types which are not objects. 57. [x].toString() Will convert [x] into a string. 58. [number].parse[numbertype]([string]) This will parse [number] into the [numbertype] with [string]. 59. ^ Return true if there is one true and one false. 60. != Not equal too. (NEQ) 61. ([condition]) ? [amount] : [var] This will be like a shortcut way to an if statement. 62. switch([variable]) This will do stuff with specific cases. (e.g. switch(hi){ case 2: (do stuff)}) 63. case [value]: This will do stuff if the case is the case. 64. break Put that when you want to leave the loop/switch; should be at end of case. 65. default [value]: This will do stuff if none of the cases in the switch statement was made. 66. for ([number]; [condition]; [operation]) This will start at [number] and then do [operation] until [condition] is met. 67. continue This will just go back to the enclosing loop before reaching other code. 68. while ([condition]) This will basically do something while [condition] is true. 69. void This means no return type. 70. return This will return something when you call it to where it was called from . 71. do { } while ([condition]) Guarantees it will execute once even if [condition] isn't met. 72. printf("%[type] stuff here bah bla", [variable here]) This will let you use [variable here] with %s being where. 73. System.out.printf([text]) Another way to print? // didn't quite get but ok then 74. [type] [returntype] [name]([parameters]) { This is a way to create a method. 75. [type][[indexes]] This will create an array with [indexes] amount of indexes; default infinite. 76. int[] something = new int[20]; This will just make an array of ints with 20 ints in it. 77. for ([object] [nameOfObject] : [arrayOfObject]) { This will iterate through all of the arrayOfObject with object in use incrementing by 1 until done. 78. [object][[1]][[2]][[3]] [name] = {[value] [value] [value] \n [value] [value] [value]} [1] is how many down in array, [2] how many accross in array, [3] how many groups 79. .length This will get how long something is, text, amount of indexes in array, etc. 80. Arrays.copy0f([array], indexes); This will copy the array and how many indexes into another array. 81. Arrays.toString([array]) Convert the whole array into one huge string. 82. Arrays.binarySearch([array], [object]) This will search for [object] in [array]. A trustable fitness App Step Set Go (Walk and Earn): [https://app.stepsetgo.com/i/abhishx637](https://www.youtube.com/redirect?q=https%3A%2F%2Fapp.stepsetgo.com%2Fi%2Fabhishx637&stzid=UgyFPVj3fS5kKsWxkId4AaABAg&redir_token=QUFFLUhqbGhOUkpSUG5RdEtwcjBFM2ZLOVpnaHNjM3FYZ3xBQ3Jtc0tueWgxNDlXczFlOEkwUFN4N1VYY0Nxa0VlUngzT0k2SUEzYWljNk9nbVpuWGxTX1cwLUV3RW1Db3dJSWNyQXo5QTF4SVBaaXhvb0I3V3VRMXJ5SkhKbmNrQmlVbVF0MDV3dE9JbFoyNjB1TjRoel9rcw%3D%3D&event=comments) Please do hear my friend's cover video for a lil stress buster: <https://youtu.be/ZO2yI8VqUcc> If this helped you, Please do check my Eyantra Robotics competition vlog: <https://youtu.be/bXwcWLQpHg0>

Klavye keyboard Shortcuts:

{} altgr 7 ve 8

() shift 8 ve 9

> Alt062 < alt 060

[] alt 8 ve 9

ALT SHİFT F format düzgünleştiriyor

(altgr ? işareti \\\)

|| or

&& and (shift 6).

Print etme:

console.log("zamcık");

//coment//

/\* uzun comment \*/

VARIABLES:

Undefined: henüz tanımlanmamış

Null: hiçbirşey

Boolean: true/false

String:TEXT

Symbol: benzersiz

Number

Object: store pairs

Var ve let kurmak için

Const değişmez

== eşittir

!= eşit değil

=== Is identical to (equal to and of the same data type)

!== Is not identical to

> Is greater than >= Is greater than or equal to < Is less than <= Is less than or equal to

alert( "5" == 5 ); // This will alert "true". They're both "5".

alert( "5" === 5 ); /\* This will alert "false". They're both "5", but they're not the same data type. \*/

+= Adds the value to itself ++ artı + ekler

-- eksi 1 çıkarır,

Let/Var izin verilen:

username, $ ile başlayan, \_ile başlayan

izin verilmeyen:

user\_name,21sayıyla başlayan, özel karakter, let

22.56 gibi sayılara float denir

2 veya -3 gibi sayılara integers denir.

IF- ARREYS

False olanlar: false, null, undefined, 0, “”, not a number”

|| or && and (shift 6).

let age = 45

if (age<=40){

console.log("40 altında yazdır")

}else if( (age >40 ) && (age<50)){

console.log("40 ile 50 arası")}

/\*

let age = 45 iken sonuç: 40 ile 50 arası

let age = 33 iken sonuç: 40 altında yazdır

50 üstündeyse birşey yazmıyor else if (age>50) {console.log("40 üzeri")} ekleyebilirsin.

|| ekleyince sonuç değişmiyor

(50< age >40) bu şekilde de yazılır. Ikinci parantez olmasa da olur.

if( true ) {    // Do something. }

var foo = [5, "five", "5"];

if( foo[1] === "five" )

{ alert("This is the word five, written in plain English."); }

Sonuç true olmazsa kod çalışmaz.

var test = "testing";

if( test == "testing" ) {

  alert( "You haven't changed anything." );

} else {

   alert( "You've changed something!" );

}

.

var yunus = [2]

if(yunus == "1"){

    console.log(true)

    }else{console.log("hayır")}

    //sayı ile string karışık olsa bile doğru veriyor

var yunus = 7;

if( yunus > 5 ) {

  alert( "beşten büyük." );

} else {

   alert( "beşten küçük!" );

}

for (let i = 0; i < 5; i++){

    console.log("zaa")

}

//sonuc: zaa zaa zaa zaa zaa

Let I : değer veriyorsun

……………i < 5, bu değer 5 ten küçük olduğu sürece aşağıdaki kodu yaz.

i++  bir ekle. Kaç tane ekleneceğini gösteriyor.

if i ikiye bölündüğünde sonuç 0 değilse, 2'ye bölünemiyorsa yazdır (1,3,5)

for (let i = 1; i <= 5; i++){

    if (i % 2 !==0) console.log("i")

}

//sonuc: i i i

5'ten 1'e doğru 2'ye bölünemeyenleri gösteriyor

for (let i = 5; i >= 1; i--){

    if (i % 2 !==0) console.log("i")

//sonuc iii

}

listem =["elma","armut","banana"]

console.log(listem[1])

//SONUC: ARMUT

const alfabe = ['a', 'b', 'c'];

for (const zaaa of alfabe) {

  console.log(zaaa);}

// expected output: "a""b""c". for in yaparsan 0,1,2 çıkıyor.

listenin 2. Sırasındaki öğeyi print ETMEK İÇİN.

var canavarlar = {"Kanada":"geyik","Kutup":"ayı","iran":"kedi"};

for(let icerik in canavarlar){

console.log(icerik);

}

//sonuc:

Kanada

Kutup

iran

içeriğin de içeriğini almak için

var canavarlar = {"Kanada":"geyik","Kutup":"ayı","iran":"kedi"};

for(let icerik in canavarlar){

console.log(canavarlar[icerik]);

}

//sonuc hepsini cıkarır:

geyik

ayı

kedi

HER İKİSİNİ DE ALMAK İÇİN

var canavarlar = {"Kanada":"geyik","Kutup":"ayı","iran":"kedi"};

for(let icerik in canavarlar){

console.log(icerik);

console.log(canavarlar[icerik]);

}

//sonuc hepsini cıkarır:

Kanada

geyik

Kutup

ayı

iran

kedi

TOPLAMA

var ZAMCIK = 10 + 5

var ZAMCIK  = 10 - 7

var ZAMCIK = 10 \* 7

var ZAMCIK = 10 / 5

console.log(ZAMCIK)

ZAMCIK = ZAMCIK + 1;  YERINE  ZAMCIK ++ ZAMCIK--

//küsüratlı da oluyor

var ZAMCIK = 5.7 \* 2.5

sonuc: 13.25

Bölündükten sonra artan sayı printleniyor

var REMAINDER;

REMAINDER = 11 % 3

console.log(REMAINDER)

//sonuc 2

variable a artı ekleme

var a = 10;

a = a + 12; YERINE a += 12

a = a – 3; YERINE  a -= 3

a = a \* 3   YERINE  a \*= 5

a = a / 3   YERINE  a /= 5

isim de combine olur

var isim = "yunus "

var soyisim = "ince"

var tamisim = isim + soyisim

console.log(tamisim)

sonuc: yunus ince

Listeyi yazdırma

var ZAMCIK = "adım \"yunus\" zaaaa \"zamcık\"";

console.log(ZAMCIK)

//sonuc adım "yunus" zaaaa "zamcık"

var ZAMCIK = "adım \"yunus\" zaaaa \"zamcık\"";

console.log(ZAMCIK)

//sonuc adım "yunus" zaaaa "zamcık"

var ZAMCIK = '"adım""yunus" "zaa"';

console.log(ZAMCIK)

//sonuc: "adım""yunus" "zaa"

var ZAMCIK = ["adım","zamcık","zaa"]

console.log(ZAMCIK)

//sonuc [ 'adım', 'zamcık', 'zaa' ]

var ZAMCIK = ["adım","zamcık","zaa"]

for(let icerik in ZAMCIK)

console.log(ZAMCIK)

/\*sonuc:

[ 'adım', 'zamcık', 'zaa' ]

[ 'adım', 'zamcık', 'zaa' ]

[ 'adım', 'zamcık', 'zaa' ]

    \*/

   var ZAMCIK = ["adım","zamcık","zaa"]

   for(let icerik in ZAMCIK)

   console.log(ZAMCIK[icerik])

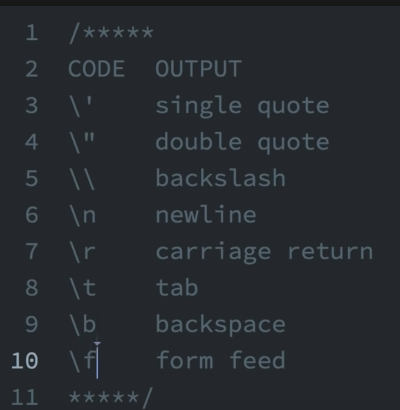
   /\*sonuc

adım

zamcık

zaa

\*/



var ZAMCIK = 'birinci ';

ZAMCIK += "ikinci";

console.log(ZAMCIK)

//sonuç: birinci ikinci

birleştirme

var isim = "yunus ";

var cumle = "merhaba " + isim + "nasılsın?"

console.log(cumle)

sonuç: merhaba yunus nasılsın?

Isim uzunluğu

var uzunluk = 0;

var isim = "yunus";

uzunluk = isim.length;

console.log(uzunluk)

//sonuc:5

ismin ilk harfi

var ilkharf = "";

var isim = "yunus";

ilkharf = isim[0];

console.log(ilkharf)

//sonuc: y

**Veya**

var isim = "yunus";

var ilkharf = isim[0];

console.log(ilkharf)

//sonuc: y

Kısaca

var ilkharf = ""; ilkharf = isim[0];

//yerine

var ilkharf = isim[0];

son harf

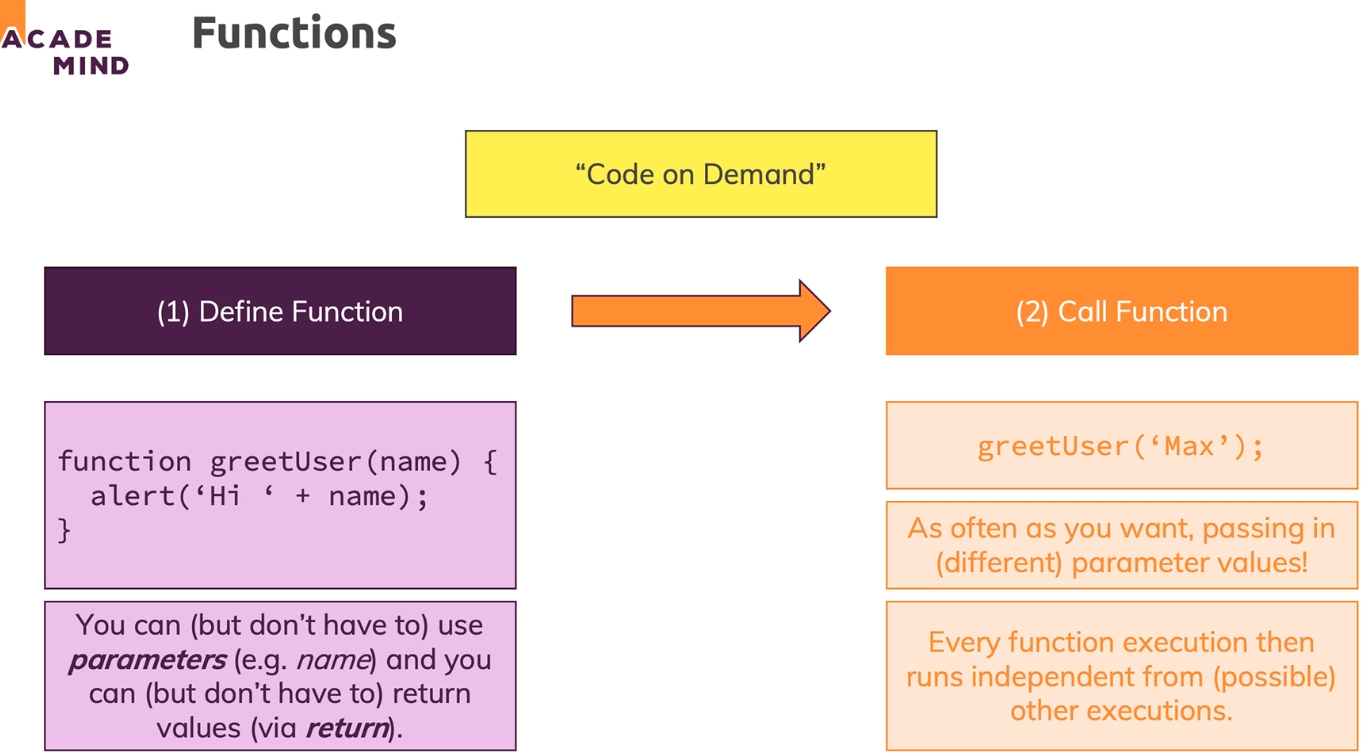
var isim = "yunus";

var sonharf = isim[isim.length - 1];

console.log(sonharf)

//sonuc s

FUNCTION



function doldur(isim, sıfat, fiil, betim) {

    var result = "gezerken " + sıfat + " " + isim + " " +"bana geldi, çok " + betim;

    return result;

  }

  console.log(doldur("köpek", "büyük", "ran", "scary"));

//sonuc:gezerken büyük köpek bana geldi, çok scary

???

ortadaki sayıyı bul

var errayim = [59,60,70];

var verim = errayim[1];

console.log(verim)

//SONUC 60

ortadaki harfi bul

var errayim = "yunus";

var verim = errayim[1];

console.log(verim)

//SONUC: U

ortasını değiştir 12 50 oldu

var dizim = [11,12,13];

dizim[1] = 50;

console.log(dizim)

//SONUC [ 11, 50, 13 ]

//sadece iceriği yazdırmak için

var dizim = [11,12,13];

dizim[1] = 50;

for(let icerik in dizim) {

console.log(dizim[icerik]);

}

…çoklu seçim

var dizim = [[11,12,13], [1,2,3], [5,6,7], [1,4,5], [5,2,1]];

var zam = dizim[1][2];

console.log(zam);

//sonuc: 3….0’dan saydığı için ikincinin üçüncüsü.

Götüne ekleme

var dizim = [[11,12,13], [1,2,3], ];

dizim.push([1],"yunus",3)

console.log(dizim);

//sonuc: [ [ 11, 12, 13 ], [ 1, 2, 3 ], [ 1 ], 'yunus', 3 ] ]

//push(buraya ne yazarsan olduğu gibi ekleniyor)

Sondan birini kaldırma pop sonu shift ilkini kaldırır.

var dizim = [1,2,3];

var kaldirilan = dizim.pop();

console.log(dizim)

console.log(kaldirilan);

//sonuc: [ 1, 2 ]   3

Başını kaldırıp başka şey ekleme

var dizim = ["yunus", "za", "kedi"];

dizim.shift();

dizim.unshift("smile")

console.log(dizim)

//sonuc: [ 'smile', 'za', 'kedi' ]

function ekle(bir,iki) {

    let sonuc = bir + iki;

    console.log("sonuç " + sonuc);

}

ekle(5,6);

ekle(8,9)

/\*sonuc:

sonuç 11

sonuç 17

function yeniden() {

    console.log("anan zaa");}

yeniden();

//sonuc: her yeniden() yazdığında anan zaaa çıkıyor.

function zammm(a,b) {

    console.log(a-b);

}

zammm(10,3);

//sonuc: 7 A,b gibi değerleri sayıya çevirir

Const değişmez, let/var değişken

function isimlik(bir,iki) {

    let tamisim = bir + " " + iki;

    console.log("sonucum: " + tamisim);

}

isimlik("yunus","ince");

//sonuc:  sonucum: yunus ince

SONRADAN FONKTION CAGIRMA

function ekle(bir,iki) {

    let sonuc = bir + iki;

    return sonuc;

}

let final= ekle(5,6);

console.log(final)

//sonradan ve hemen çağırma

function ekle(bir,iki) {

    let sonuc = bir + iki;

    console.log(sonuc)

}

ekle(5,6);

Template literals backtick (` `) ortaya formül

let a = 5;

let b = 10;

console.log('Fifteen is ' + (a + b) + ' and\nnot ' + (2 \* a + b) + '.');

// "Fifteen is 15 and

// not 20."

Bunun yerine

let a = 5;

let b = 10;

console.log(`Fifteen is ${a + b} and

not ${2 \* a + b}.`);

// "Fifteen is 15 and

// not 20."

var foo = 'bar';

console.log(`Let's meet at the ${foo}`);

// Let's meet at the bar

`string text`

`string text line 1

string text line 2`

`string text ${expression} string text`

tag`string text ${expression} string text`

let zam = 20

let result;

result = 5 + zam;

console.log(result)

//sonuc 25: diğer yazım şekli

let zam = 20

let result = 5 + zam;

console.log(result)

var items = ["foo", "bar", "baz"];

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

  console.log( items[i] );}

  //sonuc

foo

bar

baz

shift-unshift baştan kaldırır başa ekler. Pop-push sondan kaldırır sona ekler

var karakterler = ["yunus", "yinjie", "gokiburi","iruka"];

 console.log(karakterler) //sonuç: [ 'yunus', 'yinjie', 'gokiburi', 'iruka' ]

console.log(karakterler.length) //sonuç:4

console.log(karakterler[1]) //sonuç:yinjie

//index length'ten 1 düşük olur. bunu düzeltmek için

console.log(karakterler[karakterler.length-1])  //sonuç:iruka

karakterler.push("ince")

karakterler.push("sauron")

console.log(karakterler)

//push götüne ekler. Sonuç: [ 'yunus', 'yinjie', 'gokiburi', 'iruka', 'ince', 'sauron' ]

karakterler.pop()

karakterler.pop()

console.log(karakterler)

//pop her zaman sondakileri siler. sonuç: [ 'yunus', 'yinjie', 'gokiburi', 'iruka' ]

//shift baştan kaldırır, unshift başa ekler

karakterler.shift() //yunusu siler. ilk baştakini siler

karakterler.unshift("samurai")

karakterler.unshift("ninja")

console.log(karakterler)

//[ 'ninja', 'samurai', 'yinjie', 'gokiburi', 'iruka' ]

var fruits = ["Banana", "Orange", "Apple", "Mango"];  
var x = fruits.pop();      // the value of x is "Mango"

var fruits = ["Banana", "Orange", "Apple", "Mango"];

var x = fruits.push("Kiwi");   //  the value of x is 5

console.log(x)

fruits.unshift("Lemon");    // Returns 5

Kaldırılanı gösterme

var karakterler = ["yunus", "yinjie", "gokiburi","iruka"];

karakterler.unshift("bir")

console.log(karakterler)

//yukarda bir eklendi sonuç: [ 'bir', 'yunus', 'yinjie', 'gokiburi', 'iruka' ]

var kaldırılan = karakterler.shift()

console.log(kaldırılan) //kaldırılanı gösterir. sonuç: bir

ortadan silme ve ekleme splice

var karakter = ["ichi","ni","san"]

console.log(karakter)

//sonuc: [ 'ichi', 'ni', 'san' ]

karakter.splice(0,1,"İlkEklenen") //0'dan başlayıp 1 tanesini sil, "" içindekini ekle.istediğin kadar ekleyebilirsin.

console.log(karakter)

diğer örnek

var karakter = ["ichi","ni","san"]

karakter.splice(1,2,"İlkEklenen","İkinciEklenen") //1'den başlayıp 2 tanesini sil, "" içindekileri ekle.

console.log(karakter)

//sonuç: [ 'ichi', 'İlkEklenen', 'İkinciEklenen' ]

Sıralama reverse sort

Sayıları sort etmek için formül:

var points = [40, 100, 1, 5, 25, 10];

points.sort(function(a, b){return a - b});

.

var points = [40, 100, 1, 5, 25, 10];

points.sort(function(a, b){return a - b});

// now points[0] contains the lowest value

// and points[points.length-1] contains the highest value

var points = [40, 100, 1, 5, 25, 10];

points.sort(function(a, b){return b - a});

// now points[0] contains the highest value

// and points[points.length-1] contains the lowest value

Doğrudan en düşük ve yükseği bulmak için

//maks Math.max.apply(null, [1, 2, 3]) is equivalent to Math.max(1, 2, 3).

function myArrayMax(arr) {

    return Math.max.apply(null, arr);

  }

  //min

  function myArrayMin(arr) {

    return Math.min.apply(null, arr);

  }

var karakter = ["ichi","ni","san","yon","go"]

karakter.reverse()

console.log(karakter)

//soldan başa doğru sıralamayı tersine çevirdi

//sonuç: [ 'go', 'yon', 'san', 'ni', 'ichi' ]

var karakter = ["C","B","D","A","E"]

karakter.sort()

console.log(karakter)

//ALFABETİK sıraya göre sıralar sonuç: [ 'A', 'B', 'C', 'D', 'E' ]

var karakter = ["C","B","D","A","E"]

karakter.sort()

karakter.reverse()

console.log(karakter)

//ALFABETİK sıranın tersi. sonuç: [ 'E', 'D', 'C', 'B', 'A' ]

var karakter = [ 'A', 'B', 'C', 'D', 'E',]

karakter.sort()

var pozisyon = karakter.indexOf("Z")

console.log(pozisyon)

//listede olmayanlar -1 olarak gösteriliyor

var pozisyon = karakter.indexOf("B")

console.log(pozisyon)

//sonuç: 1

var pozisyon = karakter.lastIndexOf("B")

console.log(pozisyon)

//sonuç: yine 1. sondna arıyor

.array içindekini sıralama

<script>

var cars = [

  {type:"Volvo", year:2016},

  {type:"Saab", year:2001},

  {type:"BMW", year:2010}

];

displayCars();

function myFunction() {

  cars.sort(function(a, b){return a.year - b.year});

  displayCars();

}

function displayCars() {

  document.getElementById("demo").innerHTML =

  cars[0].type + " " + cars[0].year + "<br>" +

  cars[1].type + " " + cars[1].year + "<br>" +

  cars[2].type + " " + cars[2].year;

}

</script>

LOOPS

* for - loops through a block of code a number of times
* for/in - loops through the properties of an object
* for/of - loops through the values of an iterable object
* while - loops through a block of code while a specified condition is true
* do/while - also loops through a block of code while a specified condition is true

var isimler = [ 'A', 'B','C', 'D', 'E', 'F']

var isimsayısı = isimler.length;

for (var i=0; i<isimsayısı; i++)

console.log(isimler[i])

//sonuç: A B C D E F

//DİĞER YAZILIŞ ŞEKLİ i=i+2 dersen formül her 2. isimleri gösterir.a,c,e vb.

Diğer yazılışı önerilmiyor aşağıdaki

var isimler = [ 'A', 'B','C', 'D', 'E', 'F']

var isimsayısı = isimler.length;

var i=0;

for (; i<isimsayısı; i++)

console.log(isimler[i])

baştan aşağı tersten sıralama

var isimler = [ 'A', 'B','C', 'D', 'E', 'F']

var isimsayısı = isimler.length;

for (let i=isimsayısı-1;i>=0;i--)

console.log(isimler[i]);

//sonuç: FEDCBA

.

var dog = {"name":"woody","type":"dog"};

var cat = {"name":"bob","type":"cat"};

console.log(dog);

console.log(cat);

//sonuç: { name: 'woody', type: 'dog' }

//{ name: 'bob', type: 'cat' }

var petNames = ["woody","bob"];

console.log(petNames);

var pets = [dog,cat];//doğrudan kopyalayabilirsin  [{"name":"woody","type":"dog"},cat];

console.log(pets);

listeye ekleme

var dog = {"name":"woody","type":"dog"};

var cat = {"name":"bob","type":"cat"};

var petNames = ["woody","bob"];

var pets = [dog,cat];

pets.push({"name":"Roxy", "type":"dog"})

console.log(pets);

/\* sonuç: push ile dog,cat'ten sonraya bir tane daha eklendi.

[ { name: 'woody', type: 'dog' },

{ name: 'bob', type: 'cat' },

{ name: 'Roxy', type: 'dog' } ]

\*/

Console.log’u değiştirirsen herhangi birini seçip yazdırabilirsin.

console.log(pets[2]);

/\*

listeden 3. şeyi çıkarır. sonuç: { name: 'Roxy', type: 'dog' }

\*/

.

var dog = {"name":"woody","type":"dog"};

var cat = {"name":"bob","type":"cat"};

var petNames = ["woody","bob"];

var pets = [dog,cat];

pets.push({"name":"Roxy", "type":"dog"})

cat.age = 2; //hem cat.age = 2; hem de dog["age"] = 2 şeklinde yazılabilir

dog["age"] = 14;

console.log(pets);

/\* sonuç:

[ { name: 'woody', type: 'dog', age: 14 },

{ name: 'bob', type: 'cat', age: 2 },

{ name: 'Roxy', type: 'dog' } ]

\*/

Yaş özelliğini başa eklemek istersen:

cat.age=2

pets[0].age = 15 //başa eklesin

pets[2].age = 99 //üçüncüye eklesin

console.log(pets);

/\* sonuç üçünde de yaş eklendi:

[ { name: 'woody', type: 'dog', age: 15 },

  { name: 'bob', type: 'cat', age: 2 },

  { name: 'Roxy', type: 'dog', age: 99 } ]

Nested loops ???

for(let i=0;i<10;i++){

    console.log(i)}

    //sonuç: 0123456789

.

for(let i=0;i<10;i++){

    for(let k=0;k<10;k++ )

console.log(i)}

//sonuç: onar tane 0000,11111 şeklinde

.

for(let i=0;i<10;i++){

    for(let k=0;k<10;k++ )

console.log(k)}

//sonuç: on tane 0123456789

.

for(let i=0;i<10;i++){

    for(let k=0;k<10;k++ )

console.log(i,k)}

//sonuç: on tane

0 0

0 1

0 2

0 3

0 4

0 5

0 6

0 7

0 8

0 9

1 0

.

FOR EACH

const numbers = [1, 2, 3, 4, 5];

numbers.forEach(function(zaaa) {

    console.log(zaaa);

});

//sonuc:

1

2

3

4

5

Diğer yazılışı

Arrow Function Representation

numbers.forEach(zaaa => console.log(zaaa));

.

numbers.forEach((zaaa, endeks) => {

    console.log('endeks: ' + endeks + ' değer: ' + zaaa);

});

sonuc: endeks ve değer optional

endeks: 0 değer: 1

endeks: 1 değer: 2

endeks: 2 değer: 3

endeks: 3 değer: 4

endeks: 4 değer: 5

.

numbers.forEach((zaaa, endeks, dizi) => {

    console.log(dizi);

});

sonuc:

[ 1, 2, 3, 4, 5 ]

[ 1, 2, 3, 4, 5 ]

[ 1, 2, 3, 4, 5 ]

[ 1, 2, 3, 4, 5 ]

.

.    [1,2,3,4,5,6,7],

    [8,10,5,7,3,22,6,42],

    [123,54,12,11,9,15]];

    let bigHero = {characters: [

    {name:"hiro",voice:"ryan"},

    {name:"baymax",voice:"scot"},

    {name:"tamago",voice:"jamie"},

    {name:"fred",voice:"tj"}

    ]};

    let chars = bigHero.characters;

    for(let i=0,len=chars.length;i<len;i++){

    //yukarı aynı aşağı farklı

    {console.log(chars[i])}

/\*//bigHero characters içindekileri sıralıyor sonuç

{ name: 'hiro', voice: 'ryan' }

{ name: 'baymax', voice: 'scot' }

{ name: 'tamago', voice: 'jamie' }

{ name: 'fred', voice: 'tj' }

\*/

console.log(chars[i].name)}

/\* sonuç:

hiro

baymax

tamago

fred

\*/

console.log(chars[i].["voice"])}

/\* sonuç:

ryan

scot

jamie

tj

diğer örnekler

var numbers =[

    [1,2,3],

    [4,5,6],

    [7,8,9],

    0

];

    //Aşağısı değişecek

console.log(numbers[0])

//sonuç: [ 1, 2, 3 ] birinci grubun tamamı

console.log(numbers[0][0])

//sonuç:1. birinci grubun birinci öğesi

for(var i=0; i<numbers.length;i++){

    for(var j=0;j<numbers[i].length;j++)

console.log(numbers[i][j]);

}

//sonuç: 123456789

Loop seçenekleri

. var index;

var a = ["a", "b", "c"];

for (index = 0; index < a.length; ++index) {

    console.log(a[index]);

}

//alttaki alternatif

a.forEach(function(entry) {

    console.log(entry);

});

//alttaki alternatif

var index, len;

var a = ["a", "b", "c"];

for (index = 0, len = a.length; index < len; ++index) {

    console.log(a[index]);

}

}

//alttaki alternatif sonuclar hep abc

.

let dwarf = ["ali", "veli","zeli","deli"];

dwarf.forEach(function(öğe,endeks,dizin) {

    console.log(endeks,öğe);

if(öğe === "veli"){

öğe = öğe.toUpperCase();

}

else {

    öğe = öğe.toLowerCase();

}

console.log(endeks,öğe);

});

//Veli adı büyük harf kalanı küçük harf olsun. dizin silinebilir.

.

let ranks = ['A', 'B', 'C'];

for (let i = 0; i < ranks.length; i++) {

    console.log(ranks[i]);

}

Code language: JavaScript (javascript)

//Output:

A

B

C

Yerine kısaca ???

let ranks = ['A', 'B', 'C'];

ranks.forEach(function (e) {

    console.log(e);

});

MAP INDEX

let dwarf = ["ali", "veli","zeli","deli"];

let isimuzunlugu = dwarf.map(function(öğe,endeks,dizin){

return öğe

});

console.log(isimuzunlugu)

//return yerine endeks yazarsan endeksi gösteriyor

// öğe yazarsan isimleri azıyor.

let dwarf = ["ali", "veli","zeli","deli"];

let isimuzunlugu = dwarf.map(function(öğe,endeks,dizin){

let uzunluk =öğe.length

return uzunluk;

});

console.log(isimuzunlugu)

//öğelerin kaç harften oluştuğu yazıyor [ 3, 4, 4, 4 ]

Diğer kısaca formülü

let dwarf = ["ali", "veli","zeli","deli"];

let isimuzunlugu = dwarf.map(function(öğe,endeks,dizin){

return öğe.length

});

console.log(isimuzunlugu)

daha da sadece hali

let dwarf = ["ali", "veli","zeli","deli"];

let zaaa = dwarf.map(function(öğe){

return öğe

});

console.log(zaaa)

İÇERİKLERİ BİRLEŞTİRİP SIRALAMA

const ZAAA = ['Fire', 'Air', 'Water'];

console.log(ZAAA.join());

// expected output: "Fire,Air,Water"

console.log(ZAAA.join(''));

// expected output: "FireAirWater"

console.log(ZAAA.join('-'));

// expected output: "Fire-Air-Water"

var a = ['Wind', 'Water', 'Fire'];

a.join();      // 'Wind,Water,Fire'

a.join(', ');  // 'Wind, Water, Fire'

a.join(' + '); // 'Wind + Water + Fire'

a.join('');    // 'WindWaterFire'

KELİME VEYA HARF HARF AYIRMA

var a = "bir,iki,üç,dört";

var b = a.split();

console.log(b)

//[ 'bir,iki,üç,dört' ] kelimeler ayrılıyor

var a = "bir,iki,üç,dört";

var b = a.split(“”);

console.log(b)

//alfabeler ayrılıyor.[ 'b', 'i', 'r', ',', 'i', 'k', 'i'

const ZAAA = 'Hello World. How are you doing?'

const splits = ZAAA.split(' ', 3)

console.log(splits)

//This script displays the following:

//["Hello", "World.", "How"]

var fruits = ["Banana", "Orange", "Apple", "Mango"];

var energy = fruits.join();

console.log(energy)

//Banana,Orange,Apple,Mango. erray string'e dönüştü.

var fruits = ["Banana", "Orange", "Apple", "Mango"];

var energy = fruits.join(" and ");

console.log(energy)

//Banana and Orange and Apple and Mango

var meyve = ["Banana", "Orange", "Apple", "Mango"];

var b = meyve.join();

console.log(b)

//Banana,Orange,Apple,Mango

var c = b.split()

console.log(c)

//[ 'Banana,Orange,Apple,Mango' ]

DİĞER ÖRNEKLER

let a =[2,5,6,7,8]

let b = [2,3,1]

let c=a.concat(b)

console.log(c)

//iki arrey'in içeriğini doğrudan birleştiriyor, içerik+erray yapmıyor

let d=[].concat(a,b).sort()

console.log(d)

//içerikleri birleştirip sıralıyor. 11234567 diye. veya abcdef diye.

.

Split join

//string to arrey  dizim.split()

//arrey to string  öğe.join()

let sentence = "merhaba adım zamcık zaa";

let words = sentence.split("");

console.log(words)

//split. tüm harfleri tek tek ayırdı "m""e""r" diye. join tam tersi.

let words = sentence.split("").sort();

//ayrıca alfabetik sıraladı.

let çizgili = words.join("-");

console.log(çizgili)

// - - -a-a-a-a-a-a-b-c-d-e-h-k-m-m-m-r-z-z-ı-ı

let oneword = words.join("");

console.log(oneword);

//aaaaaabcdehkmmmrzzıı

let x = sentence.split("").join("");

console.log(x)

//sonuc: merhaba adım zamcık zaa

let x = sentence.split("").join("-");

console.log(x)

//sonuc: m-e-r-h-a-b-a- -a-d-ı-m- -z-a-m-c-ı-k- -z-a-a

.

let sentence = "merhaba adım zamcık zaa";

let words = sentence.split(" ");

console.log(words)

//[ 'merhaba', 'adım', 'zamcık', 'zaa' ] kelime kelime ayırdı. Parantez içi boşluğu unutma.boşluk olmazsa alfabe alfabe ayırıyor

m',

'e',

'r',

'h',

Every some

let numbers = [12,14,14,35,45,546,56]

let yirmiuzeri = numbers.every(function(num){

    return num>20;

});

console.log(yirmiuzeri)

//false çıktı: 20 altında sayı var diye.

let onuzeri = numbers.every(function(num){

     console.log(num)

     //kolsolda bu sayıları göster dedik.araya formül giriyor

    return num>10;

    });

console.log(onuzeri)

//true çıktı, 10 üzerinde hepsi,

Array includes method. Içinde \*\* varsa true yoksa false verir.

let avrat = ['joe', 'jane', 'mary']

console.log(avrat.includes('jane')

[1, 2, 3].includes(2)

true

.

console.log(['joe', 'jane', 'mary'].includes('jane')); //true

.

Filter sayılar

var numbers = [1, 3, 6, 8, 11];

var lucky = numbers.filter(function(zaaa) {

  return zaaa > 7;

});

console.log(lucky)

//sonuc [ 8, 11 ]

.

Filter, isimler

var heroes = [

    {name: "batman", franchise: "dc"},

    {name: "ironman", franchise: "marvel"},

];

var marvelHeroes =  heroes.filter(function(hero) {

    return hero.franchise == "marvel";});

console.log(marvelHeroes)

//[ { name: 'ironman', franchise: 'marvel' } ]

TÜM YAZILIŞ ŞEKİLLERİ FOR LOOP, ARROW, MAP,FILTER

var kızlar = [

    {name: "jenny", franchise: "akuma", age: 18 },

    {name: "cardi", franchise: "summoner", age: 21}];

//yukarısı sabit

var sikilecek = kızlar.filter(function(amcık){

    return amcık.age == 18

})

console.log(sikilecek)

//kısaca arrow ile

var sikilecek = kızlar.filter(amcık => amcık.age == 18);

console.log(sikilecek)

**//LOOP ile**

let sikilecek = [];

for (let i = 0; i<kızlar.length;i++){

    if (kızlar[i].age<19){

        sikilecek.push(kızlar[i]);

    }

}

console.log(sikilecek)

.

DİĞER ŞEHİR ÖRNEĞİ

let cities = [

    {name: 'Los Angeles', population: 3792621},

    {name: 'New York', population: 8175133},

    {name: 'Chicago', population: 2695598},

    {name: 'Houston', population: 2099451},

    {name: 'Philadelphia', population: 1526006}

];

.

let bigCities = [];

for (let i = 0; i < cities.length; i++) {

    if (cities[i].population > 3000000) {

        bigCities.push(cities[i]);

    }

}

console.log(bigCities);

/\*

[

  { name: 'Los Angeles', population: 3792621 },

  { name: 'New York', population: 8175133 }

]

\*/

Filter ile

let bigCities = cities.filter(function (e) {

    return e.population > 3000000;

});

console.log(bigCities);

arrow ile

let bigCities = cities.filter(city => city.population > 3000000);

console.log(bigCities);

.

For example, the following illustrates how to chain the three methods: filter(),sort(), and map():

cities

    .filter(city => city.population < 3000000)

    .sort((c1, c2) => c1.population - c2.population)

    .map(city => console.log(city.name + ':' + city.population));

Code language: JavaScript (javascript)

Output:

Philadelphia:1526006

Houston:2099451

Chicago:2695598

Code language: CSS (css)

How it works.

    First, the filter() method returns the cities whose populations are less than 3 million.

    Second, the sort() method sorts the resulting cities by the populations in descending order

    Third, the map() method show each element in the result array in the web console

.

REDUCE

var euros = [29.76, 41.85, 46.5];

var sum = euros.reduce( function(total, amount){

return total + amount

});

console.log(sum)

// sum   118.11

Arrow yazılışı

const euros = [29.76, 41.85, 46.5];

const sum = euros.reduce((total, amount) => total + amount);

console.log(sum)

// sum   118.11

. const euros = [29.76, 41.85, 46.5];

const average = euros.reduce((total, amount, index, array) => {

  total += amount;

  if( index === array.length-1) {

    return total/array.length;

  }else {

    return total;

  }

});

console.log(average)

//ortalamasını alıyor

.

var x = 5,

y = 10

var result = ""

result = x+y

console.log(result)

//var result = x+y diye kısaca yazılabilir

.

DİĞER KANAL

**JavaScript Event Listeners**

<div class="grandparent">grandparent

    <div class="parent">parent

        <div class="child">child</div>

    </div>

</div>

Yeni eklenen divler hariç

const divs=document.querySelectorAll("div")

divs.forEach(div=>{

    div.addEventListener("click",

    ()=>{

        console.log("hi")

    })

})

///tüm divler için "hi" yazar.

tüm divlerde “hi” yazsın. Yeni eklenenler de dahil.

document.addEventListener("click", e=> {

    if (e.target.matches("div")){

        console.log(hi)

    }

}) //bu tüm divlere tıkladığında işe yarar. bunu function yapmak için aşağıdakini kullan:

function evrenselListener(type,selector,callback){

    document.addEventListener(type,e=>{

        if(e.target.matches(selector)) callback(e)

    })

}

evrenselListener("click","div",e=>{console.log("hi")})

Buuble içten dışa, capture dıştan içe doğru. Buraya tıklayınca üçü birden açılıyor.

RENK DEĞİŞTİRME BUTONU

div class="colorbutton"><button>Change color</button></div>

const btn = document.querySelector("div.colorbutton");

function random(number) {

  return Math.floor(Math.random() \* (number+1));

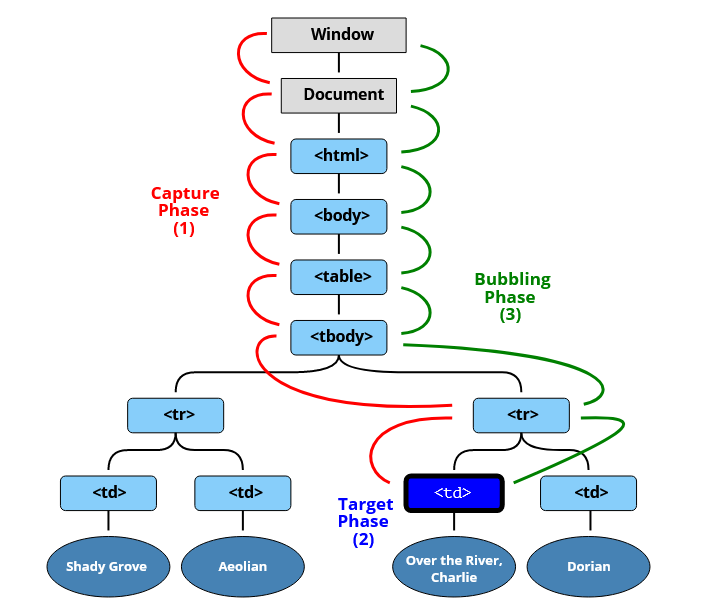
}

btn.onclick = function() {

  const rndCol = 'rgb(' + random(255) + ',' + random(255) + ',' + random(255) + ')';

  document.body.style.backgroundColor = rndCol;

}



2 saniye sonra kodu kaldır.

setTimeout(() => {

    parent.removeEventListener"click",()=>{console.log("hi")}

}, 2000);

**DOM HTML**

Javaya Yeni bir kutu ekledik

const yenidiv = document.createElement("div")

yenidiv.style.width ="200px"

yenidiv.style.height ="200px"

yenidiv.style.backgroundColor="purple"

document.body.append(yenidiv)

**yeni paragraph ekleme**

var paragraf = document.createElement("p");

var yazı = document.createTextNode("This is new.");

paragraf.appendChild(yazı);

var yeni = document.getElementById("div1");

yeni.appendChild(paragraf);

//son child değil ilk eklemek istiyorsan son yazıyı şöyle değiştir.

yeni.insertBefore(paragraf,child);

**. Javaya Yeni bir kutu ekledik**

const yenidiv = document.createElement("div")

yenidiv.style.width ="200px"

yenidiv.style.height ="200px"

yenidiv.style.backgroundColor="purple"

document.body.append(yenidiv)

**kaldırmak için**

var elmnt = document.getElementById("p1");

elmnt.remove();

For browsers that does not support the remove() method, you have to find the parent node to remove an element:

var parent = document.getElementById("div1");

var child = document.getElementById("p1");

parent.removeChild(child);

html içinde tarih butonu sabit

<div class="tarih"><button onclick="document.getElementById('demo').innerHTML = Date()">The time is?</button></div>

Html içinde kaybolan buton. Buton adı yok fonksiyonu devam ediyor.

<button onclick="this.innerHTML = Date()">The time is?</button>

Sıl normal js format

Html:

<div class="tarih"> <button onclick="displayDate()">The time is?</button> </div>

js

function displayDate() {

    document.getElementById("demo").innerHTML = Date();

  }

.

Escape button

|  |  |  |
| --- | --- | --- |
| \' | ' | Single quote |
| \" | " | Double quote |
| \\ | \ | Backslash |

var x = "We are the so-called \"Vikings\" from the north.";

var x = 'It\'s alright.';

\b Backspace

\f Form Feed

\n New Line. Alt satır

\r Carriage Return

\t Horizontal Tabulator

\v Vertical Tabulator

"abracadabra".**lastIndexOf**('ab'); // return 7

"abracadabra".**indexOf**('ab') // return 0

str.indexOf('k',4) // 11 (it start search from 4th position)

## slice(şuradan başla, şuraya kadar)

var str = "Apple, Banana, Kiwi";

var res = str.slice(7,13);

//sonuç: Banana

var str = "Apple, Banana, Kiwi";

eksiyse tersten başlar

var res = str.slice(-12, -6);

//sonuç: Banana

//tek sayı yazarsan oradan itibaren sonuna kadar

var str = "Apple, Banana, Kiwi";

var res = str.slice(7);

//sonuç: Banana, Kiwi

substring

substring //aynı sadece negatif yazılmıyor

var str = "Apple, Banana, Kiwi";

var res = str.substring(7, 13);

//sonuç: Banana,

var str = "Apple, Banana, Kiwi";

var res = str.substr(-4);

//The result :Kiwi

Replace

<div class="tryit"><button onclick="myFunction()">Ee sonra?</button></div>

<p id="deneme"> ananın amına çam diktim!</p>

function myFunction() {

  var str = document.getElementById("deneme").innerHTML;

  var txt = str.replace("ananın amına çam diktim!"," topla şimdi kozalakları");

  document.getElementById("deneme").innerHTML = txt;

}

To replace case insensitive, use a **regular expression** with an /i flag (insensitive):

str = "Please visit Microsoft!";

var n = str.replace(/MICROSOFT/i, "W3Schools");

global değiştirmek için

str = "Please visit Microsoft and Microsoft!";

var n = str.replace(/Microsoft/g, "W3Schools");

sonuç: //Please visit W3Schools and W3Schools!

var text1 = "Hello World!";       // String  
var text2 = text1.toUpperCase();  // text2 is text1 converted to upper

var text1 = "Hello World!";       // String  
var text2 = text1.toLowerCase();  // text2 is text1 converted to lower

he concat() method can be used instead of the plus operator. These two lines do the same:

### Example

var text = "Hello" + " " + "World!";  
var text = "Hello".concat(" ", "World!");

The trim() method removes whitespace from both sides of a string:

var str = "       Hello World!        ";  
alert(str.trim());

padStart and padEnd to support padding at the beginning and at the end of a string.

let str = "5";

str = str.padStart(4,0);

// result is 0005

let str = "5";

str = str.padEnd(4,0);

// result is 5000

## toString()

var x = 123;  
x.toString();            // returns 123 from variable x  
(123).toString();        // returns 123 from literal 123  
(100 + 23).toString();   // returns 123 from expression 100 + 23

## toFixed() yazılan sayı kadar ortalamasını alıyor.

## toPrecision() yine benzer

var x = 9.656;  
x.toFixed(0);           // returns 10  
x.toFixed(2);           // returns 9.66  
x.toFixed(4);           // returns 9.6560  
x.toFixed(6);           // returns 9.656000

sayıya çevirme

|  |  |
| --- | --- |
| Number() | Returns a number, converted from its argument. |
| parseFloat() | Parses its argument and returns a floating point number |
| parseInt() | Parses its argument and returns an integer |

Number(true);          // returns 1  
Number(false);         // returns 0

parseInt("10");         // returns 10  
parseInt("10.33");      // returns 10  
parseInt("10 20 30");   // returns 10  
parseInt("10 years");   // returns 10  
parseInt("years 10");   // returns NaN

parseFloat("10.33");     // returns 10.33

var last = fruits[fruits.length - 1]; son öğeyi çıkarır

erray I liste yapma

eski yöntem

function makeList() {

    // Establish the array which acts as a data source for the list

    let listData = [

        'Blue',

        'Red',

        'White',

        'Green',

        'Black',

        'Orange'

    ],

    // Make a container element for the list

    listContainer = document.createElement('div'),

    // Make the list

    listElement = document.createElement('ul'),

    // Set up a loop that goes through the items in listItems one at a time

    numberOfListItems = listData.length,

    listItem,

    i;

    // Add it to the page

    document.getElementsByTagName('body')[0].appendChild(listContainer);

    listContainer.appendChild(listElement);

    for (i = 0; i < numberOfListItems; ++i) {

        // create an item for each one

        listItem = document.createElement('li');

        // Add the item text

        listItem.innerHTML = listData[i];

        // Add listItem to the listElement

        listElement.appendChild(listItem);

    }

}

// Usage

makeList();

diğer ???

var fruits, text;

fruits = ["Banana", "Orange", "Apple", "Mango"];

text = "<ul>";

fruits.forEach(myFunction);

text += "</ul>";

document.getElementById("demo").innerHTML = text;

function myFunction(value) {

  text += "<li>" + value + "</li>";

}

fruits[fruits.length] = "Lemon";    // adds a new element (Lemon) to fruits

fruits.push("Lemon");    // adds a new element (Lemon) to fruits

araya bosluk koyarsan undef:ned oluyor

Array.isArray(fruits);   // returns true

toString() ile join() aynı sayılır, fruits.join(" \* ");ayıracı ayarlayabilirsin.

Html içine ekleme butonu splice

<button onclick="myFunction()">Try it</button>

<p id="demo1"></p>

<p id="demo2"></p>

<script>

var fruits = ["Banana", "Orange", "Apple", "Mango"];

document.getElementById("demo1").innerHTML = "Original Array:<br>" + fruits;

function myFunction() {

  fruits.splice(2, 0, "Lemon", "Kiwi");

  document.getElementById("demo2").innerHTML = "New Array:<br>" + fruits;

}

</script>

The first parameter (2) defines the position **where** new elements should be **added** (spliced in).

The second parameter (0) defines **how many** elements should be **removed**.

fruits.splice(fruits.length-1, 1);  sondan birini siler.

var myChildren = arr1.concat(arr2, arr3);   // Concatenates arr1 with arr2 and arr3

var myChildren = arr1.concat("Peter");

var fruits = ["Banana", "Orange", "Apple", "Mango","zam"];

var kesilen = fruits.slice(3);

console.log(kesilen)

//sonuç:mango,zam

var citrus = fruits.slice(1,3); birinciden başla üçüncüye kadar kes.

Resim boyutunu değiştirme: ??????????

Html

<img id="kedi" src="../resimmm/kedi.jpg">

Js

document.getElementById("myImg").width = "500";

tıklayınca resimin değişmesi

html

    <img id="kedicik" src="../resimmm/kedi.jpg">

Js

/\* global document \*/

var dogImage = document.getElementById('kedicik');

dogImage.onclick = function(){

var myImages = dogImage.getAttribute('src');

if(myImages === '../resimmm/kedi.jpg') {

    dogImage.setAttribute('src', '../resimmm/12.PNG');

}else{

    dogImage.setAttribute('src', '../resimmm/kedi.jpg');}}

. hoşgeldin username

var nameButton =document.querySelector("button");

var myheading = document.querySelector("h1");

function setUserName(){

    'use strict';

    var myName=window.prompt("adını yaz");

    localStorage.setItem("name",myName);

    myheading.textContent="iyi günler, " + myName;

}

if (!localStorage.getItem("name")){

    setUserName();

}else{

        var storedName = localStorage.getItem("name");

        myheading.textContent= "iyi günler, " + storedName;

    }

    nameButton.onclick=function(){

        "use strict";

        setUserName();

    };

Html:

<h1>"have a nice day"</h1>

<button class="button">butoniki</button>

## **addEventListener**

.tıklayınca zaa yazsın

buton.onclick = function(){

    alert('zaaa')

}

. mouse hover

buton.onmouseover = function(){

    alert('zaaa')

}