Arrays(Diziler)

Bir javascript dizisi üç farklı yöntemle tanımlanabilmektedir.

Example

//1.Condensed Array

let fruits = new Array("Apple", "Cherry", "Orange");

//2.Literal Array

let numbers = [1, 2, 3];

//3.Regular Array

let days = new Array();

days[0] = "Sunday";

days[1] = "Monday";

//Hata meydana gelmez. 2, 3 ve 4. index'ler boş olarak tanımlanır. Bu yüzden dizi

//6 elemanlı olur.

days[5] = "Friday";

console.log(days); //[ "Sunday", "Monday", <3 empty slots>, "Friday" ]

console.log(days.length); //6

* Çok boyutlu diziler tanımlayabilirsiniz.

Example

let arr1 = Array(new Array(), new Array(), new Array());

let arr2 = [[], [], []];

let arr3 = new Array();

arr3[0] = new Array();

arr3[1] = new Array();

arr3[2] = new Array();

Array Sınıfı

Diziler üzerinde işlemler yapılırken Array sınıfının özellik ve metodları kullanılmaktadır.

Example

//Özellik ve metodlar konsol penceresinde görüntülenebilmektedir.

console.log(new Array());

length

Dizinin eleman sayısını almak ya da değiştirmek için kullanılır.

Example

let numbers = [1, 2, 3];

console.log(numbers); //[ 1, 2, 3 ]

console.log(numbers.length); //3

//Dizinin eleman sayısını değiştirdik. Bazı elemanlar silindi.

numbers.length = 2;

console.log(numbers); //[ 1, 2 ]

console.log(numbers.length); //2

push()

Dizinin sonuna eleman eklemek için kullanılır.

Example

let numbers = [1, 2, 3];

numbers.push(4, 5, 6);

console.log(numbers); //[ 1, 2, 3, 4, 5, 6 ]

unshift()

Dizinin başına eleman eklemek için kullanılır. Kullanımı push() metodu ile aynıdır.

pop()

Dizinin son elemanını verir ve bu elemanı dizi içinden siler.

Example

let numbers = [1, 2, 3];

console.log(numbers.pop()); //3

console.log(numbers); //[ 1, 2 ]

shift()

Dizinin ilk elemanını verir ve bu elemanı dizi içinden siler. Kullanımı pop() metodu ile aynıdır.

toString()

Dizi içeriğini string olarak döndürür.

Example

let numbers = [1, 2, 3];

let str = numbers.toString();

console.log(str); //1,2,3

console.log(typeof str); //string

join()

Parametre olarak verilen değeri dizi elemanları arasına ekler ve string olarak döndürür. Parametresiz kullanımda elemanlar arasına virgül eklenir.

Example

let numbers = [1, 2, 3];

console.log(numbers.join()); //1,2,3

console.log(numbers.join("+")); //1+2+3

console.log(numbers.join("--")); //1--2--3

concat()

Dizileri birleştirip geri döndürür.

Example

let arr1 = [1, 2, 3];

let arr2 = [4, 5, 6];

let arr3 = [7, 8, 9];

console.log(arr1.concat(arr2, arr3)); //[ 1, 2, 3, 4, 5, 6, 7, 8, 9 ]

reverse()

Dizi elemanlarını tersten sıralar.

Example

let numbers = [1, 2, 3];

numbers.reverse();

console.log(numbers); //[ 3, 2, 1 ]

indexOf()

Parametre olarak verilen değeri dizi içerisinde arar ve ilk geçtiği yerin index numarasını döndürür. Eşleşme olmazsa -1 değeri döner. İkinci parametrenin kullanımı isteğe bağlıdır, başlangıç index’ini belirtmek için kullanılır.

Example

let numbers = [1, 9, 7, 3, 1];

console.log(numbers.indexOf(1)); //0

console.log(numbers.indexOf(2)); //-1

console.log(numbers.indexOf(1, 1)); //4

lastIndexOf()

Parametre olarak verilen değeri dizi içerisinde arar ve son geçtiği yerin index numarasını döndürür. Eşleşme olmazsa -1 değeri döner. İkinci parametrenin kullanımı isteğe bağlıdır, bitiş index’ini belirtmek için kullanılır.

Example

let numbers = [1, 9, 7, 3, 1];

console.log(numbers.lastIndexOf(1)); //4

console.log(numbers.lastIndexOf(1, 3)); //0

console.log(numbers.lastIndexOf(1, 4)); //4

slice()

Dizinin bir bölümünü(ya da tamamını) kopyalar ve yeni bir dizi olarak döndürür. İlk parametre başlangıç, ikinci parametre bitiş index’idir. Bitiş index’i belirtilmezse dizinin sonuna kadar seçilir. Bitiş index’i negatif değer aldığında dizi sonundan bitiş index’i kadar eleman seçilmez.

Example

let numbers = [1, 2, 3, 4, 5, 6];

console.log(numbers.slice()); //[1, 2, 3, 4, 5, 6]

console.log(numbers.slice(1)); //[2, 3, 4, 5, 6]

console.log(numbers.slice(1, 4)); //[2, 3, 4]

console.log(numbers.slice(0, -2)); //[1, 2, 3, 4]

splice()

Dizi içinden eleman silmek ya da yeni elemanlar eklemek için kullanılır.

Example

//Elemanları silerken yeni elemanlar eklemek.

let arr = [1, 2, 3, 4, 5, 6];

//splice(startIndex, count, newItems)

//splice(başlangıç indeksi, çıkarılacak eleman sayısı, eklenecek elemanlar)

let deletedItems = arr.splice(2, 2, "A", "B", "C");

console.log(deletedItems); //[3, 4]

console.log(arr); //[1, 2, "A", "B", "C", 5, 6]

Example

//Eleman eklemek.

let arr = [1, 2, 3, 4];

arr.splice(2, 0, "A", "B", "C");

console.log(arr); //[1, 2, "A", "B", "C", 3, 4]

Example

//Eleman silmek.

let arr = [1, 2, 3, 4, 5, 6];

arr.splice(2, 2);

console.log(arr); //[ 1, 2, 5, 6 ]

map()

Dizi elemanlarını belirtilen fonksiyondan geçirerek dizi halinde geri döndürür.

Example

let numbers = [2, 4, 6, 8];

console.log(numbers.map(function(item){ //[ 4, 16, 36, 64 ]

return item \* item;

}));

//İkinci ve üçüncü parametrelerin kullanımı isteğe bağlıdır.

console.log(numbers.map(function(item, index, arr){ //[ 4, 16, 36, 64 ]

return arr[index] \* item;

}));

//Fonksiyon bloğundan sonra kendi parametrenizi gönderebilirsiniz.

//Parametrenize fonksiyon içerisinden this operatörü ile ulaşabilirsiniz.

//NOT : Parametre gönderilmezse this undefined olur.

console.log(numbers.map(function(item){ //[ "4-", "16-", "36-", "64-" ]

return item \* item + this;

}, "-"));

filter()

Dizi elemanlarını filtreleyip dizi halinde geri döndürür.

Example

let numbers = [2, 4, 6, 8];

console.log(numbers.filter(function(item){ //[ 2, 4 ]

return item < 5;

}));

every()

Dizi elemanlarını belirli bir kontrolden geçirir, tüm elemanlar koşulu sağlıyorsa true aksi halde false döner.

Example

let numbers = [2, 4, 6, 8];

console.log(numbers.every(function(item){ //true

return item <= 8;

}));

console.log(numbers.every(function(item){ //false

return item <= 6;

}));

some()

every() gibi çalışır fakat elemanlardan en az biri koşulu sağlıyorsa true hiçbir eleman koşulu sağlamıyorsa false döner.

Example

let numbers = [2, 4, 6, 8];

console.log(numbers.some(function(item){ //true

return item <= 2;

}));

console.log(numbers.some(function(item){ //false

return item > 8;

}));

forEach()

Dizideki her eleman için bir fonksiyon çalıştırır.

Example

let numbers = [2, 4, 6, 8];

numbers.forEach(function(item){

console.log(item); //2, 4, 6 ve 8 değerleri sırasıyla konsola yazılır.

});

|  |
| --- |
| **NOT**  filter(), some(), every() ve forEach() metodları da map() gibi iki ya da üç parametreli olarak kullanılabilmektedir. Ayrıca fonksiyon bloğundan sonra özel parametre gönderme imkanımız vardır. |