

# JS - Sıkça Kullanılan Metotlar

## String Methods

Özellik	Açıklama
<code>.concat("birinci" , "ikinci");</code>	String ifadeleri birleştirir.
<code>.length</code>	Karakter sayısını gösterir
<code>.toUpperCase();</code>	Karakterleri büyük harfe çevirir
<code>.toLowerCase();</code>	Karakterleri küçük harfe çevirir
<code>.indexOf('')</code>	Aranan ifadenin kaçınca eleman olduğunu gösterir (kelime aranırca kelimenin başladığı index baz alınır)
<code>.substring(3,5);</code>	Belirtilen aralıktaki ifadeleri getirmek için kullanılır
<code>.slice(3,5);</code>	Belirtilen aralıktaki ifadeleri getirmek için kullanılır
<code>.replace('degistirilecek_yer, yeni_kelime')</code>	Bir ifadenin değiştirilmesi için kullanılır
<code>.trim();</code>	Boşlukları kaldırır.
<code>.split(",")</code>	Virgüllerin kullanıldığı yerlerden dizi şeklinde ayırır. Boşluk veya başka şeyden de ayrılabilir. (Madde haline getirir)

## Number Methods

String İfadeyi Sayıya Çevirmek	Açıklama
<code>Number("0");</code>	İçerisindeki değeri tam sayıya çevirir.
<code>parseInt("0");</code>	İçerisindeki değeri tam sayıya çevirir.
<code>parseFloat("10.5");</code>	İçerisindeki değeri ondalıklı değere çevirir.
<code>isNaN ("10");</code>	Sayısal bir değeri olup olmadığını söyler. Sayısal değeri değilse 'True' döner.

Sayının Basamaklarını Almak	Açıklama
<code>.toFixed(5);</code>	Sayıdaki ilk 5 rakamı alır.
<code>.toFixed(3);</code>	Virgülden sonra belirtildiği kadar basamak alır.

Math Methods	Açıklama
<code>Math.PI;</code>	Pi sayısını getirir
<code>Math.round(2.4);</code>	Belirtilen sayıyı en yakınına yuvarlar (2)
<code>Math.ceil(2.4)</code>	Belirtilen sayıyı <u>yukarıya</u> yuvarlar (3)
<code>Math.floor(2.7)</code>	Belirtilen sayıyı <u>aşağıya</u> yuvarlar (2)
<code>Math.sqrt(64)</code>	Karekökü almak için kullanılır
<code>Math.pow(2,4)</code>	Solda belirtilen rakamın sağdaki kadar üssünü alır. (2.2.2.2)
<code>Math.abs(-100)</code>	Mutlak değerini alır
<code>Math.min(1,2,3,4,5,6)</code>	Belirtilen sayılar arasındaki en küçük sayıyı getirir
<code>Math.max(1,2,3,4,5,6)</code>	Belirtilen sayılar arasındaki en büyük sayıyı getirir
<code>Math.random();</code>	0-1 arasında sayı çağırmak için kullanılır
<code>Math.random()*10;</code>	0 ile 10 arasında sayı çağırmak için kullanılır (10 dahil olması için +1 eklenebilir. <code>Math.random()*10+1;</code>

## Date & Times Methods

GET Methods (Değer çağırmak için kullanılanlar)	Açıklama
<code>getDate();</code>	Ayın kaçınıcı günü olduğunu gösterir
<code>getDay();</code>	Haftanın kaçınıcı günü olduğunu gösterir (Dizi olduğu için 0'dan başlar)
<code>getFullYear();</code>	Yıl bilgisi gelir
<code>getHours();</code>	Saat bilgisi verilir
<code>getMonth();</code>	Ay bilgisi gelir (Dizi olduğu için 0'dan başlar)
<code>getMinutes();</code>	Dakika bilgisi verilir
<code>getSeconds();</code>	Saniye bilgisi verilir
SET Methods (Değer değiştirmek için kullanılanlar)	Açıklama
<code>setFullYear();</code>	Yıl değerini değiştirmek için kullanılır
<code>setMonth();</code>	Ay değerini değiştirmek için kullanılır
<code>setDate();</code>	Tarih bilgisini değiştirmek için kullanılır (ayın kaçınıcı günüyse o değişir )
<code>setMinutes();</code>	Dakika bilgisi değiştirmek için kullanılır
<code>setSeconds();</code>	

## Array Methods

Array Methods	Açıklama
<code>.push();</code>	En sona eleman eklemek için kullanılır
<code>.unshift();</code>	En başa eleman eklemek için kullanılır
<code>.pop();</code>	Dizinin son elemanını silmek için kullanılır
<code>.shift();</code>	Dizinin ilk elemanını silmek için kullanılır
<code>.indexOf('');</code>	Arama işlemini yapar ve başladığı index numarasını gösterir.
<code>.reverse();</code>	Diziyi ters çevirmek için kullanılır
<code>.sort();</code>	Sayısal veya alfabetik olarak sıralar ( <i>Varsayılan: Artan</i> )
<code>.concat();</code>	Dizileri birleřtirmek için kullanılır
<code>.splice();</code>	Dizinin belirtilen elemandan sonra kaç elemanın silineceğini ve bu işlemten sonra ( <i>istenilirse</i> ) eleman eklenmesini sağlar. <i>Örneğin:</i> <code>names.splice(0,1,"newName");</code>
<code>.find(koşul);</code>	Parantez içine koşul yazılır ( <i>bu bir fonksiyon olabilir</i> ). Belirtilen koşulu sağlayan, dizi içerisindeki <b>ilk eleman</b> çağrılır.
<code>.filter(koşul);</code>	Parantez içine koşul yazılır ( <i>bu bir fonksiyon olabilir</i> ). Belirtilen koşulu sağlayan, dizi içerisindeki <b>tüm elemanlar</b> çağrılır.