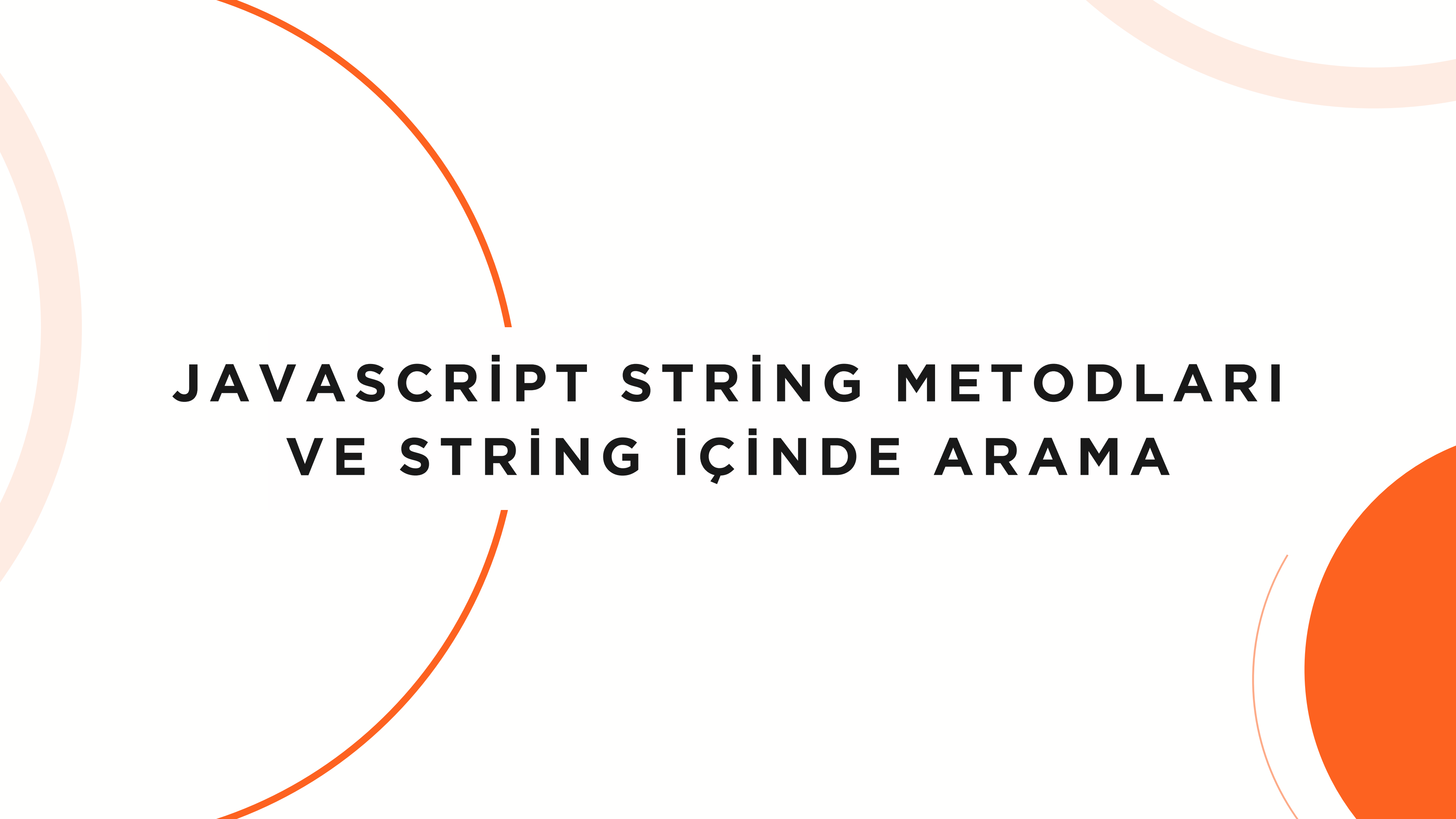




JAVASCRIPT STRING METODLARI VE STRING İÇİNDE ARAMA

1.String Metodları:

- String manipülasyonu ve analizi için kullanılan yerleşik fonksiyonlardır.
- Sadece arama yapmak için değil, aynı zamanda stringi değiştirmek, bölmek, birleştirmek vb. işlemler için de kullanılır.
- Örnek metodlar: slice(), concat(), replace(), split(), toUpperCase(), toLowerCase() vb.

2.String İçinde Arama Yapmak:

- String metodlarının bir alt kümesi olan arama işlemleri için kullanılan metodları ifade eder.
- Bir string içinde belirli bir alt dizeyi veya deseni bulmaya odaklanır.
- Örnek metodlar: indexOf(), lastIndexOf(), includes(), search(), match(), startsWith(), endsWith().



1

2

3

4

5

6

7

8

9

String Nedir ?

- Bir string, karakterlerden oluşan bir dizidir.
- Örnek: "Hello, World!"
- Çift Tırnak (""), Tek Tırnak (''), ve Backtick (`) ile string oluşturabiliriz.



1

2

3

4

5

6

7

8

9

Const- Let-Var

Const

Java Script'te "Const" kullanıldığında tanımlanan değer değiştirilemez. Eğer bir değeri değiştirmek istiyorsak "Var" veya "Let" kullanabiliriz.

- "Let":Değişkenin değerinin değişebileceğini belirtir.
- "Var":Geleneksel olarak kullanılan bir değişken tanımlama anahtar kelimesidir.Ancak modern Java Script'te "let" tercih edilmektedir.

```
const pi = 3.14;  
pi = 3,14159; // hata! Bir const degiskeninin degeri degistirilemez  
  
let pi = 3.14;  
pi = 3.14159; // gecerli cünkü "let" kullanıldı.  
  
var x = 5;  
x = 10; // gecerli "var" kullanımı.
```

String Methodları

1

2

3

4

5

6

7

8

9

1.Length: Bir stringin karakter sayısını döndürür.

```
let str = "Hello World";  
  
// length  
console.log(str.length); // 11
```

2.charAt(index) veya [index]: Belirtilen dizideki karakteri döndürür.

```
let str = "Hello World";  
  
// charAt() veya [index]  
console.log(str.charAt(0)); // "H"  
console.log(str[1]); // "e"
```

1

2

3

4

5

6

7

8

9

3.indexOf(searchValue, fromIndex): Belirli bir değeri arar ve ilk bulunan dizinini döndürür. Aramaya isteğe bağlı olarak bir başlangıç dizininden başlanabilir.

```
let str = "Hello World";

// indexOf()
console.log(str.indexOf("o")); // 4
console.log(str.indexOf("o", 5)); // 7
```

4.substring (startIndex, endIndex): Belirtilen dizinler arasındaki karakterleri içeren bir alt dize oluşturur. endIndex'in kullanımı isteğe bağlıdır.

```
let str = "Hello World";

// substring()
console.log(str.substring(6)); // "World"
console.log(str.substring(0, 5)); // "Hello"
```

5.toLowerCase() ve toUpperCase():Bir stringi büyük harfe veya küçük harfe dönüştürür.

```
let str = "Hello World";

// toLowerCase() ve toUpperCase()
console.log(str.toLowerCase()); // "hello"
console.log(str.toUpperCase()); // "HELLO"
```

6.replace (searchValue, replaceValue):Belirli bir değeri arar ve bulunan her ögeyi bir başka değerle değiştirir.

```
let str = "Hello World";

// replace()
console.log(str.replace("World", "Universe")); // "Hello Universe"
```

1

2

3

4

5

6

7

8

9

7.trim():Bir stringin başındaki ve sonundaki boşlukları kaldırır.

```
let str = "Hello World";

// trim()
let strWithSpace = "  Hello  ";
console.log(strWithSpace.trim()); // "Hello"
```

8.split(separator):Bir stringi belirten ayırıcıya göre böler ve alt dizelerden oluşan bir dizi döndürür.

```
let str = "Hello World";

// split()
console.log(str.split(" ")); // ["Hello", "World"]
```

1

2

3

4

5

6

7

8

9

9.startsWith(searchString, position) ve endsWith(searchString, length):
Bir stringin belirli bir değerle başlayıp başlamadığını veya belirli bir değerle bitip bitmediğini kontrol eder.

```
let str = "Hello World";

// startsWith() ve endsWith()
console.log(str.startsWith("Hello")); // true
console.log(str.endsWith("World")); // true
```

10.includes(searchString, position): Bir stringin belirli bir değeri içerip içermediğini kontrol eder.

```
let str = "Hello World";

// includes()
console.log(str.includes("World")); // trueA
```

11.slice(startIndex, endIndex): Belirtilen dizinler arasındaki karakterleri içeren bir alt dize oluşturur.endIndex'in kullanımı isteğe bağlıdır.

```
let str = "Hello World";

//slice()
console.log(str.slice(0, 5)); // "Hello"
```

12.padStart(targetLength, padString) ve padEnd(targetLength, padString): Bir stringi belirli bir uzunluğa ulaşacak şekilde belirli bir karakterle başa veya sona doldurur.

```
let str = "Hello";

//padStart ve padEnd
let paddedStr = str.padStart(10, ".");
console.log(paddedStr); // ".....Hello"

let paddedSr = str.padEnd(10, "!");
console.log(paddedStr); // "Hello!!!!!"
```



1

2

3

4

5

6

7

8

9

DİNLEDİĞİNİZ İÇİN TEŞEKKÜRLER