

concat () : Yöntemi çağırarak dizileri birleştirir ve yeni bir dizi döndürür.

```
let kelime = 'Ufuk ' ;  
  
let kelime2 = 'Ozkar' ;  
  
console.log(kelime.concat(' ', kelime2));  
  
Çıktı =>   Ufuk  Ozkar
```

endsWith(): Bu yöntem bir dizinin hangi kelime ile bitip bitmeyeceğini belirlemenizi sağlar. Bu yöntem büyük küçük harfe duyarlıdır. İki değer döndürür. True ya da false.

```
let kelime = 'Ufuk Ozkar';  
  
console.log(kelime.endsWith('Ozkar'));  
  
Çıktı =>   true
```

startsWith(): Bu yöntemle de bir dizinin hangi kelime ile başladığını buluruz. Bu yöntem büyük küçük harfe duyarlıdır. İki değer döndürür. True ya da false.

```
let kelime = 'Ufuk Ozkar';  
  
console.log(kelime.startsWith('Ufuk'));  
  
Çıktı => true
```

includes(): Bu yöntemle bir dizinin içinde istediğimiz kelimenin olup olmadığını kontrol ederiz. True, false döndürür.

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle yapıyor. ' ;  
  
console.log(kelime.includes('bu'));  
  
Çıktı =>   true
```

indexOf (): Bu yöntemle dizinin içinde aradığımız kelimenin yerini buluruz. Karşını çıkan ilk değeri döndürür. Büyük küçük harf uyarlıdır.

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle odevi yapıyor. ';\nlet arananKelime = 'odevi';\nconsole.log(kelime.indexOf(arananKelime));\n\nÇıktı => 14
```

lastIndexOf():Yöntem Bu yöntemle dizinin içinde aradığımız kelimenin yerini buluruz. Karşını çıkan son değeri döndürür. Büyük küçük harf uyarlıdır.

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle odevi yapıyor. ';\nlet arananKelime = 'odevi';\nconsole.log(kelime.lastIndexOf(arananKelime));\n\nÇıktı => 31
```

padEnd(): Bu yöntemle diziye dolgu uygulayıp dizinin belirli bir uzunluğa getiririz.

```
let kelime = 'Ufuk Ozkar ';\n\nconsole.log(kelime.padEnd(30, '*'));\n\nÇıktı => Ufuk Ozkar *****
```

padStart(): Bu yöntemle dizinin başına dolgu uygulayıp diziyi belirli bir uzunluğa getiririz.

```
let kelime = 'Ufuk Ozkar';\n\nconsole.log(kelime.padStart(13, '*'));\n\nÇıktı => ***Ufuk Ozkar
```

repeat(): Bu yöntemle istediğimiz diziyi istediğimiz kadar kopyalayabiliriz, tekrar ettirebiliriz.

```
let kelime = 'Ufuk Ozkar ';  
  
console.log('iyi olan kim: ',kelime.repeat(3));  
  
Çıktı => iyi olan kim:  Ufuk Ozkar Ufuk Ozkar Ufuk Ozkar
```

replace () :Bu yöntemle dizide istediğimiz kelimeyi istediğimiz kelimeyle değiştirebiliriz. Aradığımız kelimedenden birden fazla sayıda varsa hepsini birden de değiştirebiliriz.

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle odevi yapıyor. ';  
  
let degisecekKelime = 'odevi';  
  
console.log(kelime.replace(degisecekKelime,'gorevi'));  
  
Çıktı => Ufuk Ozkar bu gorevi titizlikle odevi yapıyor.
```

Global kullanılırsa ;

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle odevi yapıyor. ';  
  
let degisecekKelime = /odevi/gi;  
console.log(kelime.replace(degisecekKelime,'gorevi'));  
  
Çıktı => Ufuk Ozkar bu gorevi titizlikle gorevi yapıyor.
```

split(): Bu yöntemle dizideki istediğimiz sıradaki karakterin ya da kelimenin ne olduğunu öğreniriz.

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle odevi yapıyor. ';  
  
let karakter = kelime.split ('');  
  
console.log(karakter[3]);  
  
Çıktı => k
```

Tırnaklar arasına boşluk bırakılırsa karakter değil kelimeyi alıyor.

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle odevi yapıyor. ';  
let kelime2 = kelime.split (' ');  
  
console.log(kelime2[3]);  
Çıktı => odevi
```

slice():Bu yöntemle dizinin bir bölümünü seçip yeni bir dizi döndürürüz. Negatif değerler verebiliriz. Negatif değer verdiğimizde sondan saymaya başlar. İkinci bir değer girmediğimiz takdirde dizi sonuna kadar seçer.

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle odevi yapıyor. ';\n\nlet kelime2 = kelime.slice(10,19);\n\nconsole.log(kelime2);\n\nÇıktı => bu odevi
```

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle odevi yapıyor. ';\n\nlet kelime2 = kelime.slice(-15);\n\nconsole.log(kelime2);\n\nÇıktı => odevi yapıyor.
```

substr():Bu yöntem belirtilen bir indeksten başlayıp istediğimiz uzunluktaki kısmı döndürüyor.

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle odevi yapıyor. ';\n\nlet kelime2 = kelime.substr(5,5);\n\nconsole.log(kelime2);\n\nÇıktı => Ozkar
```

substring():Bu yöntem belirtilen bir indeksten başlayıp istediğimiz indekse kadar olan kısmı döndürüyor. İkinci indeks verilmezse dizi sonuna kadar alır.

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle odevi yapıyor. ';\n\nlet kelime2 = kelime.substring(5,10);\n\nconsole.log(kelime2);\n\nÇıktı => Ozkar
```

```
let kelime = 'Ufuk Ozkar bu odevi titizlikle odevi yapıyor. ';\nlet kelime2 = kelime.substring(5);\nconsole.log(kelime2);\n\nÇıktı => Ozkar bu odevi titizlikle odevi yapıyor.
```

trim(): Bu yöntem dizinin her iki ucundaki boşlukları kaldırır.

```
let kelime = '      Ufuk Ozkar bu odevi titizlikle yapıyor.      ';\nconsole.log(kelime.trim());\n\nÇıktı => 'Ufuk Ozkar bu odevi titizlikle yapıyor.'
```

trimEnd(): Bu yöntem dizinin her sonundaki boşlukları kaldırır.

```
let kelime = '      Ufuk Ozkar bu odevi titizlikle yapıyor.      ';\nconsole.log(kelime.trimEnd());\n\nÇıktı => '      Ufuk Ozkar bu odevi titizlikle yapıyor.'
```

trimStart(): Bu yöntem dizinin her başındaki boşlukları kaldırır.

```
let kelime = '      Ufuk Ozkar bu odevi titizlikle yapıyor.      ';\nconsole.log(kelime.trimStart());\n\nÇıktı=> 'Ufuk Ozkar bu odevi titizlikle yapıyor.      '
```