



BATCH : B45-46-71  
LESSON : **Javascript**  
DATE : 11.04.2022  
SUBJECT : **Advanced JS**



 [techproeducation.com](https://techproeducation.com)

 [info@techproeducation.com](mailto:info@techproeducation.com)

 +1 (917) 768-7466



## Web Storage API

Kullanıcı tarafında data saklayabilmek için HTML5 ile birlikte gelen Storage API kullanılır.

`localStorage`

Veri, kullanıcı veya app silene kadar saklanır.

`sessionStorage`

Veri, oturum bazlı saklanır. Tarayıcı kapatıldığında veri silinir.

HTML5 ten önce kullanıcı tarafında veri saklamanın tek yolu cookie kullanmaktı. Ancak cookie ler hem kullanıcı iznine tabi hem de veri saklama sınırı düşük olduğu için artık Storage API kullanılmaktadır. Cooki lerde sınır 4KB iken, Web Storage ile en az 5MB bilgi saklanabilir. Web storage, domain ve protokol bazlı çalışmaktadır. Aynı domain ve protokole sahip dosyalar aynı origin içindedir. Web storage e aynı origin içindeki dosyalardan erişilebilir. Ancak yine de hassas dataların (eposta, şifre, kredi kartı vs.) burada saklanması yanlıştır.



## Web Storage API

localStorage ve sessionStorage de data manipölasyonu için kullanılan methodlar

setItem

Veri saklamak için

getItem

Veriyi almak için

removeItem

Veriyi silmek için

```
localStorage.setItem("counter", 1);
```



## Web Storage API

PRACTISE

Sayfa her ziyaret edildiğinde, ziyaretçiye siteye kaçınıcı kez girdiğini söyleyen programı yazınız.



## Web Storage API

PRACTISE

Bir form üzerinden kullanıcının ismin alıp, bundan sonraki siteye girişlerinde ismi ile «merhaba» yazan programı yapınız.



## Rest & Spread Operator

ES  
6



Rest ve Spread operatörleri 3 nokta ile ifade edilirler ancak birbirinden farklıdır.

REST

Farklı değişkenlerden gelen verileri bir dizi içinde toplar

SPREAD

Dizi veya object içindeki verileri ayrı değişkenlere açar.



## Rest Operator

**Örnek:** Toplama işlemi yapan bir fonksiyon yapınız. Ancak bu fonksiyona farklı miktarda sayı parametre olarak gönderilebilmelidir.

```
function topla(...sayilar){  
    return sayilar.reduce( (toplam, x)=> toplam + x);  
}  
  
console.log(topla(3,5,6));  
console.log(topla(13,5,16,3,56,76,1));
```

Gelen tüm parametreleri sayılar isimli dizi içinde alacaktır.



## Rest Operator

**Örnek:** Öğrencinin adını ve soyadı ile belirsiz sayıda notu parametre olarak alan ve notların ortalamasını bulup, öğrenci adı ile birlikte ortalamayı döndüren fonksiyonu yazınız.

```
function ortalama(ad, soyad, ...notlar){  
  let ort = notlar.reduce( (toplam, x)=> toplam + x);  
  return `${ad} ${soyad} ortalaması: ${ort/notlar.length}`  
}  
  
console.log(ortalama("Ali", "Gel", 3, 5, 6));  
console.log(topla(("Ayşe", "Koş", 13, 5, 16, 3, 56, 76, 1)));
```

İlk iki parametre haricinde geri kalan tüm parametreleri bir dizi içine alacaktır