

Adım 1: Veri Yapılarını Oluşturma

1. **User Objesi:** İçinde id, username, email ve password alanları olan bir **user** objesi oluşturun.
2. **Products Dizisi:** En az 4 tane obje içeren bir **products** dizisi oluşturun. Her objenin id, name ve price özellikleri olsun.

Adım 2: Dizi İşlemleri (Map & Filter)

3. **Map ile Listeleme:** map metodunu kullanarak yukarıda oluşturduğunuz tüm products dizisindeki product'ların sadece name'lerini konsola yazdırın bir **arrow function** yazın.
4. **Filter ile Ayıklama:** Price değeri belirli bir değerin (örneğin 100 TL) üzerinde olan product'ları filtreleyip yeni bir diziye atayan ve o dizidekifiltrelenmiş productların yine map ile sadece name'lerini konsola yazan bir fonksiyon yazın.

Adım 3: İleri Seviye İşlemler (Reduce & Spread & Rest)

5. **Reduce ile Toplam:** Bir önceki adımda filtrelediğiniz ürünlerin toplam fiyatını **reduce** kullanarak hesaplayın ve konsola yazdırın.
6. **Spread ile Ekleme:** Orijinal products dizisini bozmadan, **spread** operatörünü kullanarak dizinin sonuna yeni bir ürün ekleyin ve oluşan yeni dizideki ürün isimlerini yine map ile tekrar konsola basın.
7. **Rest ile Ayırma:** En başta oluşturduğunuz user nesnesinden password alanını **rest** operatörüyle ayırin. Geri kalan tüm kullanıcı bilgilerini (...others) tek bir nesne olarak konsola yazdırın.

Adım 4: React Bileşenlerinde Gelişmiş Destructuring

8 . Destructuring (Parçalama): Aşağıdaki product objesini probsdan alıp parçalayan ve h1 h2 h3 taglarında frontend için gösteren bir reacy komponent yapınız.

Komponent de gösterilecek alanlar : title , likeCount, price, seller.nickName, seller.rating

Bu gösterilecek alanlara göre destructuring yapınız.

```
{  
  id: 3,  
  title: "Decibel Dominator",  
  kullanımDurumu: "Yeni",  
  likeCount: 200,  
  hype: 8120,  
  price: 270,  
  imageUrl: "/products/c.webp",  
  isOpenToTeklif : true,  
  isSaticiOder: true,  
  seller: {  
    id: 56,  
    nickName: "SesUzmanı",  
    imageUrl: "/customer/c3.png",  
    rating: 8.9,  
    followingCustomer: 5211,  
    rozetler: [  
      {  
        title: "7/24 Destek",  
        imageUrl: "/customer/destekRozet.png"  
      }  
    ]  
  }  
}
```

ÖRNEK

API'den aşağıdaki gibi user adında bir nesne geldiğini düşünelim.

- **Veri Yapısı:**

```
const user = {
  id: 1,
  profile: {
    firstName: "Ahmet",
    lastName: "Yılmaz",
    details: {
      age: 22,
      university: "Erciyes Üniversitesi"
    }
  }
};
```

Bu API' dan gelen user nesnesini probs (parametre) olarak alan ve komponentin içinde işleyip frontend de gösteren komponent aşağıdaki gibidir.

```
const UserCard = (data) => {

  // BURADA COMPLEX DESTRUCTURING YAPILACAK
  const { profile: { firstName, details: { age, university } } } = data

  return (
    <div>
      <h1>Öğrenci: ${firstName}</h1>
      <h2>Yaş: ${age}</h2>
      <h3>Okul: ${university}</h3>
    </div>
  )
}
```

Bu parçalamayı direk probsdan alırken de yapabildik ki bu tarz parçalama daha sık kullanılıyor.

```
const UserCard = ({ profile: { firstName, details: { age, university } } }) => {

  return (
    <div>
      <h1>Öğrenci: ${firstName}</h1>
      <h2>Yaş: ${age}</h2>
      <h3>Okul: ${university}</h3>
    </div>
  )
}
```