

JS-9320P-1



Hedefimiz

Belirtilen veri setindeki Star Wars karakterlerinin görsel olarak listelendiği bir web sayfası yapalım. Veri setimiz:

<https://github.com/WildCodeSchool/starwars-api/blob/master/db.json>

Koşullar ve bilgilendirmeler

- Veri setini alıp oluşturduğumuz JavaScript dosyası içerisinde bir değişkene tanımlayalım
- Öncelikle aşağıdaki adımları sırayla okuyarak kafamızda canlandırmaya çalışalım.
- Sayfamız açıldığı sırada karakterler LİSTELENMEMİŞ olarak başlamalı, yani ekranda herhangi bir karakter gözükmemeli. Yeşil renkli bir “Karakterleri Göster” butonuna bastığımızda tüm karakterler listelenmeli. Butonun hemen altında oluşturacağımız bir Bootstrap flex yapısı (row, col) içerisinde listeleyebiliriz

- Sayfa ilk açıldığında ilk başta kartlara dair herhangi bir html kodu olmamalı. BUTONA BASILDIKTAN SONRA kartlara ait html kodları javascript aracılığıyla oluşturulmalı
 - Yani “display: none” koyup kaldırarak vs. değil direkt JavaScript aracılığıyla sıfırdan HTML kodu oluşturarak bu kısmı halletmeliyiz
- Tüm karakterleri göstermek için TEK bir buton koyacağız, her karakter için ayrı bir buton olmayacak. O tek butona bastığımızda tüm karakterler toplu bir şekilde gösterilecek
- Butona bastığımızda karakterlerin kartları JavaScript sayesinde HTML kodu oluşturularak ekrana yansıtılmalı. Yani butona basana kadar o kartlara dair html kodu yokken butona bastığımız anda kartların html kodları oluşmalı
- Karakterler gösteriliyorken butonun içerisindeki yazı “Karakterleri Gizle” olmak üzere değişmeli, rengi de kırmızı olmalı. O butona tıklayınca karakterler kartları silinmeli, buton tekrar “Karakterleri Göster” butonuna dönüşmeli
- Şimdi kurguyu kafamızda baştan oluşturmaya çalışalım:
 - Elimizde verilerin olduğu bir array var:
 - Bu array'deki her bir Star Wars karakteri için:
 - bir Bootstrap card oluşturup:
 - bu karakterlerin ismini, resmini ve nereli olduklarını (homeworld) bu card'ların içinde göstereceğiz
 - Bunları elimizle HTML'e girmemeliyiz, JavaScript ile veri setinden oluşturmalıyız. Yani amacımız, kartları JavaScript ile oluşturmak
- Array'in içerisindeki her bir eleman (karakter) için tekrarlayan belirli işlemler gerçekleştireceğiz
 - Aynı işlemleri bir listedeki tüm elemanlarda sırayla teker teker gerçekleştirmek için JavaScript'te ne kullanıyorduk?
 - Algoritmanın bu kısmını kafamızda kurmaya çalışalım:
 - Array'deki her bir eleman için belirli işlemler gerçekleştireceğiz
 - Gerçekleştirdiğimiz işlem, birkaç HTML elementi oluşturmak olacak

- Bu durumda JavaScript kullanarak HTML elementi oluşturmayı öğrenmeliyiz. Bunun için yardımcı kaynaklar kısmına bakabiliriz
- Card'ı kendimiz yazmak yerine bootstrap ile oluşturmak tasarım yükümüzü fazlasıyla hafifletecektir
- Yukarıdaki adımları hallettikten sonra bir adım daha ekleyeceğiz: Butona tekrar bastığımızda karakterleri HTML'den tamamen silelim
 - Buton aç kapa şeklinde çalışabilmeli, karakterleri istediğimiz kadar gösterip gizleyebilmeliyiz yani
 - Butonu değiştirmek zorluyorsa ilk başta iki tane buton koyabiliriz. Bir buton karakterleri göstermeye yararken diğer buton da karakterleri gizlemeye yarayabilir. Bu butonları koyup işlevsellik sağladıktan sonra nasıl tek seferde tek buton göstereceğimizi düşünüp ona göre bir çözüm üretebiliriz
 - Yeter ki ilk önce bir şeyleri ana hatlarıyla çalışır hale getirelim yani, daha sonra nasıl daha iyi çalışacak hale getirebileceğimizi düşünürüz zaten

Adımlar



İzlememiz gereken adımlar aslında basit arkadaşlar. Sırasıyla bu adımları özetleyelim:

1. GitHub linkindeki veri setini alıp bir JavaScript dosyası içerisinde değişkene atayarak kayıt edelim
2. Butona basıldığında bir işlem gerçekleşecek, yani buton'a click yapıldığında (onclick) bir fonksiyon tetiklenecek. Bu fonksiyonu tanımlayalım. İsmi "renderCharacters" olabilir çünkü yaptığımız işlem, karakterleri HTML sayfamızda göstermek (yani render etmek) olacak
3. "renderCharacters" fonksiyonunda yaptığımız şey, ilk önce JavaScript ile butonun altında bir div oluşturup ona row class'ı vermek olacak
4. Şimdi de bu row'un içine karakter kartlarının HTML kodunu eklememiz lazım:
 - a. Bunun için ilk önce HTML tarafında, JavaScript'ten bağımsız bir şekilde (yani JavaScript kullanmadan, sadece HTML kodu yazarak) elimizle bir kart örneği yapalım

- b. HTML CSS ile güzel bir dizayn örneği oluşturduktan sonra yaptığımız bu örneğin kodlarını şimdi JavaScript tarafına kopyalayarak “template literal string” içerisinde bir değişkende tutalım. Ardından, HTML tarafındaki (.html dosyasındaki) kart kodunu silelim çünkü kart prototipimizin dizaynını HTML tarafında tamamladık, bundan sonrası JavaScript’te devam edecek
 - c. Şimdi de elimizdeki karakterler array’ini bir döngüye sokarak her bir karakter için bu template literal string’i, daha önceden oluşturmuş olduğumuz row’un HTML kodunun içerisine ekleyelim
 - i. Yani döngüdeki her bir eleman için, row’un içerisindeki HTML koduna ilave yapacağız
 - ii. row’un içindeki HTML kodu baktığımızda string aslında, biz de bu string’e yeni karakter kodunu string olarak ilave edeceğiz. Yapmamız gereken şey bir string’e başka bir string eklemek (iki string’i toplamak toplamak) yani sadece
 - d. Oluşan görüntüyü inceleyelim. Karakter array’inde atıyorum 10 karakterimiz varsa, yukarıdaki adımları takip ettikten sonra ekranda 10 tane karakter kartı gözükecek. Fakat buradaki sorun, her karakterin şu an aynı olması, çünkü **4.b** adımında bir karakter kartı örneğinin HTML kodunu JavaScript değişkeninde tutarak **4.c** adımında da bu kodları render ettik, fakat kodda değiştirmemiz gereken karakter ismi, karakter anavatanı gibi yerleri değişken olarak ayarlamadık
 - e. Kodda değiştirmemiz gereken kısımları, döngünün içerisindeki karakterin verisine göre template literal string içerisinde değiştirelim. Template literal string’ler, bu işi kolayca yapmamızı sağlıyordu
5. Bu aşamaya kadar “renderCharacters” fonksiyonunun işlevi kartları ekrana bastırmaktı. Şimdi yapacağımız şey ise “Karakterleri Göster” butonunu, “Karakterleri Gizle” butonu olarak değiştirmek
- a. Bunun için butonun yazısını (içindeki HTML’ini), CSS kodlarını ve “onclick” fonksiyonunu JavaScript ile değiştirmemiz gerekecek
 - b. Yeni “onclick” fonksiyonumuzun ismini “removeCharacters” tarzı bir şey yapabiliriz

- c. “removeCharacters” fonksiyonunun işlevi ise, daha önceden oluşturduğumuz row class’lı div elementini tamamen kaldırmak olabilir. row class’lı div elementini kaldırıncı, içindeki karakter kartları da onunla birlikte kalkmış olacak. Böylece karakterlerimizi “gizlemiş” daha doğrusu yok etmiş olacağız
 - d. Yukarıdaki adımları takip ettikten sonra “removeCharacters” fonksiyonuna son bir ekleme daha yapmamız gerekecek: “Karakterleri Gizle” butonunu tekrar “Karakterleri Göster” butonuna çevirmek
6. Tüm bu işlemleri tamamladıktan sonra yukarıdaki koşullar ve bilgilendirmeleri tekrar okuyup eksik bir kısım var mı diye kontrol edelim

Yardımcı Kaynaklar

- <https://getbootstrap.com/docs/5.3/components/card/>
-  “js dom innerHTML”
 - Bu, bir HTML elementinin içeriğini değiştirmemizi sağlıyor. Karakter kartlarını JavaScript ile bir div’in içinde (row’da) oluşturacağız. Bunun için, karakter kartlarını kapsayan div’in içeriğini (yani innerHTML’ini) değiştirmemiz gerekecek
 - Her bir kart için aslında bu div’e (row class’ı olan div’e yani) döngü içerisinde HTML eklemesi yapacağız. Yani her kart’ın HTML kodu sırayla row’un innerHTML’ine eklenecek
-  “js dom create html element”
 - Bunu, Bootstrap card’larını içine koyacağımız row’un div’ini oluşturmak için kullanabiliriz
 - div’i oluşturduktan sonra oluşturduğumuz elemente row class’ı ekleyebiliriz
 - div’i oluşturduktan sonra zaten getElementById ile halihazırda var olan bir elementi seçmişiz gibi ilgili elementin üzerinde istediğimiz şekilde işlem yapabiliyoruz. Yani istersek class da ekleyebiliriz, istersek innerHTML’ini de değiştirebiliriz. Bir kere oluşturduktan sonra gerisi bizim ne yapmak istediğimize kalmış tamamen
 - “Karakterleri Göster” butonuna basınca bu şekilde bir row oluşturup, “Karakterleri Gizle” butonuna basınca da bu row’u tamamen kaldırabiliriz

- Karakterlerin listelendiği row'u kaldırmak için:
 - 🗝️ "js remove element"
- <https://wesbos.com/template-strings-html>
 - Kartları oluştururken bunu kullanacağız
- <https://code.mu/en/javascript/book/prime/dom/manipulation/loop-elements-creating/>
ve

<https://stackoverflow.com/questions/29758498/using-a-for-loop-within-innerhtml>

- Bu kaynaklardaki örneklerde toplu bir şekilde nasıl element oluşturabileceğimiz gösterilmiş. İki yöntem de birbirine yakın çözümler sunuyor, ikisini de kullanabiliriz. Yukarıdaki "Adımlar" kısmında ikinci kaynaktaki çözümlere benzer bir çözüm anlatılıyor ama ilk kaynaktaki şekilde de çözebiliriz, tercihimize kalmış
- Eğer ilk kaynaktaki şekilde çözeceksek:
 - İlk başta bu linkteki aynı örneği kendimiz boş, temiz bir projede kullanmayı deneyelim. Biraz kurcalayarak iyice neymiş, ne değilmiş mantığını anlamaya çalışalım
 - Orada p elementleri oluşturulup içerilerine textContent ile "!" koyulmuş
 - Biz textContent yerine innerHTML kullanıp içerik olarak da daha önceden HTML olarak oluşturmuş olduğumuz bootstrap card'ı koyup, değişmesi gereken yerleri sırasıyla döngünün içerisindeki array elemanlarının (karakterlerin) değerlerinden halledebiliriz
- 🗝️ "toggle classnames of elements with js"
 - Butona class ekleyip çıkararak buton rengini değiştirebilmek için kullanabiliriz
- 🗝️ "js innerText vs innerHTML"
 - Bu bilgiyi doğrudan kullanmayacağız bu projede ama araştırıp öğrenmeye çalışsak güzel olur