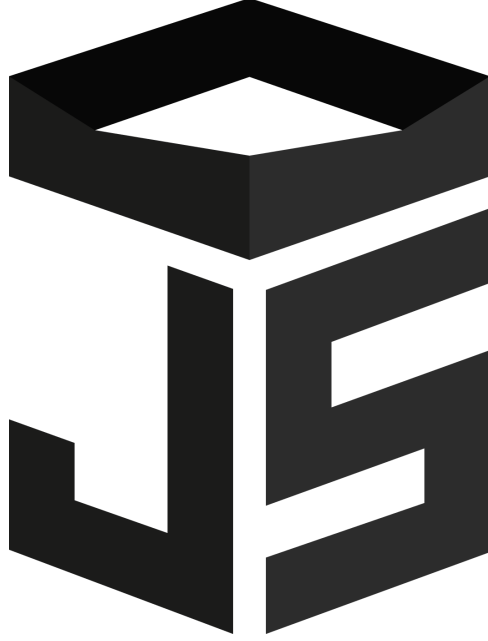


JS-2888P-2



Hedefimiz

JSON Placeholder API'si ile oluşturmuş olduğumuz kullanıcı listesine, kullanıcıların gönderilerini (post'larını) listeleme özelliği ekleyelim

Koşullar ve bilgilendirmeler

- Bir önceki ödevde, kullanıcıları listelemek için “index.html” sayfasını ve o sayfaya bağlamak için oluşturduğumuz script dosyasını kullanmıştık. Şimdi aynı proje üzerinde yeni bir HTML sayfası ve o sayfaya bağlamak için de yeni bir js dosyası oluşturacağız
 - Ödevimizin iki kısmındaki HTML ve JavaScript dosyalarımızın karışmaması için isimlendirmeleri aşağıdaki şekilde ayarlayabiliriz:
 - Önceki ödevde:
 - html dosyamızın ismi: “index.html”

- (burası projemizde ana sayfamız olarak kalabilir, yani projemizi ilk açtığımızda kullanıcılar sayfası yüklensin. o yüzden ismini “users.html” falan yapmayacağız ki “index.html” olup ana sayfa olsun, çünkü “index.html” dosyaları ana sayfa dosyası oluyordu)
- js dosyamızın ismi: “users.js”
 - (bunun eski ismi projemizin bir önceki kısmında yaptığımız şekildeydi, belki “script.js” ya da “main.js” tarzı bir şey yapmış olabiliriz, projemizi kontrol edelim, ona göre bu maddede belirtildiği gibi “users.js” ismi olmak üzere değiştirelim)
- Bu ödevde:
 - html dosyamızın ismi: “posts.html”
 - js dosyamızın ismi: “posts.js”
 - Yeni oluşturduğumuz “posts.html” sayfamıza sadece “posts.js” script’ini dahil edeceğiz, “users.js” scriptini dahil etmeyeceğiz. Yani her sayfaya sadece kendi script’i dahil edilecek, başka bir sayfanın script’i dahil edilmeyecek
- Tarayıcımızda, önceki ödevimizin arayüzünü açıp inceleyelim
 - Kullanıcı listemizi görüyoruz
 - JSON Placeholder API’sinde aslında tüm bu kullanıcıların oluşturmuş olduğu post’lar da var. Biz, bu postları kullanıcı bazlı olmak üzere filtreleyip gösterteceğiz bu ödevde
- JSON Placeholder’daki “Routes” kısmına bakalım:

Routes

All HTTP methods are supported. You can use http or https for your requests.

GET	<u>/posts</u>
GET	<u>/posts/1</u>
GET	<u>/posts/1/comments</u>
GET	<u>/comments?postId=1</u>
POST	/posts
PUT	/posts/1
PATCH	/posts/1
DELETE	/posts/1

- Burada “/comments?postId=1” kısmına baktığımızda query string kullanıldığını görüyoruz
 - Bu query string, bizim filtreleme yapabilmemizi sağlıyor
 - Yani “/comments?postId=1” endpoint’inde, tüm comment’ler (yorumlar) arasından post id’si 1 olan comment’ler filtreleniyor
 - id’si 1 olan post’a ait yorumlar gösteriliyor yani
 - Query string kısmı olmadan sadece “/comments” endpoint’ine girdiğimizde ise tüm yorumlar gösteriliyor
 - Burada tüm post’lara ait tüm comment’ler (yorumlar) gösteriliyor. Filtreleme yapılmıyor yani
 - Gördüğümüz gibi, bu API’de filtrelemeler yapabiliyoruz. Filtreleme için postId’yi kullandık
- “Routes” kısmına baktığımızda comment’lerin postId’ye göre filtrelenebildiğini gördük. Aynısını şimdi post’ları userId’ye göre filtrelemek üzere deneyelim:
 - Şu linke girelim: <https://jsonplaceholder.typicode.com/posts?userId=1>

- queryString'deki "1" kısmını sırasıyla önce "2" ile, sonra "3" ile ve son olarak "4" ile değiştirerek ayrı ayrı deneyelim
 - Gördüğümüz gibi userId filtresini değiştirdikçe bize gelen data değişiyor, yani userId'ye göre filtreleme yapabiliyoruz
- Şimdi queryString kısmını kaldıralım (linkten ? karakteri ve sonrasını silelim)
 - Gördüğümüz gibi filtre kalktı, tüm kayıtlar geldi
- JsonPlaceholder API'ye baktığımızda queryString sayesinde backend tarafında filtreleme yapıldığını, yani queryString'den gönderilen parametrelerin tespit edilip ona göre ilgili verilerin döndürüldüğünü görüyoruz
 - queryString'de gönderilen veriler backend tarafında okunmuş yani
 - Peki biz bu verileri (yani queryString parametrelerini) frontend tarafında da okuyabiliyor muyuz?
 - Mesela bizim kendi projemizde "posts.html" dosyası var. O dosyayı tarayıcımızda görüntüleyebilmek için "<http://127.0.0.1:5500/posts.html>" gibi bir adrese giriyoruz tarayıcımızda. Peki bu adresin sonuna queryString parametreleri ekleyip o parametreleri javascript'te okuyup ona göre işlem yapamaz mıyız?
 - Yani "<http://127.0.0.1:5500/posts.html>" yerine "<http://127.0.0.1:5500/posts.html?userId=1>" adresine girsek ve oradaki "?userId=1"deki "1" kısmı bizde de değişiyor olsa o değişen kısma javascript tarafında erişemez miyiz?
 - Elbette erişebiliriz. Yardımcı Kaynaklar kısmına bakalım
 - Yardımcı Kaynaklar kısmına bakıp kodu projemize yazdıktan sonra kendi projemiz "posts.html" sayfasında açıkken queryString'teki userId parametresini değiştirelim ve yardımcı kaynaklardan bulduğumuz kodun hemen altına bulduğumuz değeri console.log yapıp kontrol edelim
 - Tarayıcımızdaki adres çubuğu kısmındaki userId değeri ile console.log yaptığımızda gelen değerin aynı olması lazım
 - Elimizde tarayıcımızın adres çubuğundaki queryString kısmından gelen userId değeri var. Adresteki userId kısmını değiştirdikçe console.log yaptığımız değer de

adres çubuğuna bağlı bir şekilde değişiyor şu an, yani artık adres çubuğundaki queryString değerlerine erişebiliyoruz

- O zaman sayfamız yüklenirken JsonPlaceholder API'sine de bu console.log yaptığımız değeri gönderip kendi sitemizde filtrelemeye çalıştığımız kullanıcı gönderilerine erişemez miyiz?
 - Erişiriz elbette. Projemizdeki "posts.html" sayfasına girdiğimizde adres çubuğundaki "userId=" kısmını değiştirdikçe javascript'e yansıyan değer de ona bağlı bir şekilde değişiyor sonuçta. Biz de bu değeri alıp JsonPlaceholder API endpointine gönderebiliriz
 - Yani bizim linkimiz "<http://127.0.0.1:5500/posts.html?userId=1>" iken, <https://jsonplaceholder.typicode.com/posts?userId=1> adresine istek atıp oradaki verileri çekeceğiz. Bizdeki "1" kısmı değiştikçe JsonPlaceholder API'ye attığımız istekteki "1" kısmı da değişecek, bizimkiyle aynı olacak. Yani istek attığımız adresin sonu dinamik olacak
 - Adresi dinamik yapmak için template literal string kullanabiliriz veya direkt normal string'in sonuna + ile değişken ekleyebiliriz. Yeter ki istek attığımız adresin başı aynıyken sonu değişken olsun
- Bize gelen posts verilerini, tıpkı ödevin ilk kısmı olan kullanıcılar kısmında yaptığımız gibi bir card tasarımı oluşturup o card'lara basalım
- Bu adıma kadar geldiğimizde, "posts.html" sayfamıza girerken queryString kısmında aldığımız "userId" değerini, JsonPlaceholder API'na istek atarken kullanıp ilgili user'a ait verileri card olarak ekrana basabiliyor olduk. Fakat userId kısmı queryString'de hiç verilmezse veya o kısım boş bırakılırsa ne yapacağız?
 - Bu durumda bir prompt açıp kullanıcının bir userId girmesini isteyebiliriz
 - JsonPlaceholder API'deki "/users" kısmına baktığımızda, userId'lerin 1'den başlayıp 10'da bittiğini görüyoruz.
 - Bu durumda kullanıcıya bu adımda göstereceğimiz prompt içerisinde bir sayı almamız lazım ve aynı zamanda bu sayının 1 ile 10 arasında olması lazım
 - Farklı bir şey denk gelirse (mesela kullanıcı sayı girmek yerine yazı girerse veya girilen sayı belirtilen aralıkta olmazsa) alert ile hata

göstermeliyiz ve hata patlatıp sonraki satırlarda gerçekleşmesi beklenen tüm işlemleri iptal etmeliyiz

- “throw Error” yapmalıyız yani
- “posts.html” sayfasında işimizin çoğu bitti. Şimdi en üste bir “Kullanıcılara Dön” link butonu ekleyelim ve bu butona tıkladığımızda da kullanıcıların listelendiği ana sayfaya gitsin
- Şimdi de ana sayfadaki kullanıcı kartlarının içinde bi yere “Gönderileri görüntüle” linki koymamız lazım ve o linke tıkladığımızda projemizdeki “posts.html” sayfasına queryString ile beraber gitmesi lazım ki tıkladığımız kullanıcının post’ları karşımıza çıksın
 - user card’a link eklemek için card’ın tasarımını javascript’teki template literal string içerisinde değiştirmek zor geliyorsa tarayıcımızdan devtools’u açıp, oradan bir card elementi bulup sağ tıklayıp kopyalayarak html sayfamızda altlara bir yere yapıştırıp onun üstünde düzenlemeler yaptıktan sonra tekrar user card’ları için template literal string’i oluşturabiliriz

Yardımcı Kaynaklar

- <https://stackoverflow.com/questions/901115/how-can-i-get-query-string-values-in-javascript>
 - Query string parametrelerine erişmek için burada verilen ilk cevabı kullanabiliriz
 - Biz, örnekteki “myParam” yerine kendi kodumuzda “userId” yazmalıyız çünkü o parametreyi okumak istiyoruz
- 🗝️ “js throw error”