

**Reactı Kurmak istediğimiz dizine bu kodu yazıyoruz**

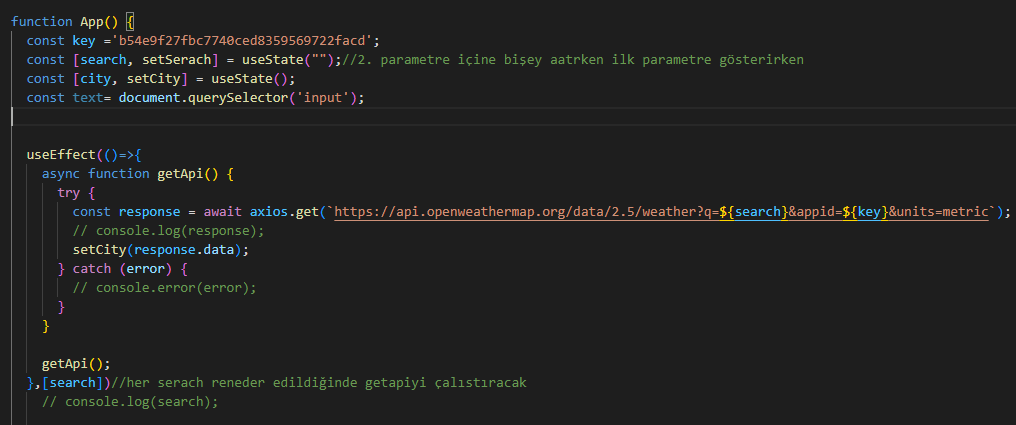
****

**Veri çekmek için stacklere yani hooklara ihtiyaç var örenek olarak axios u kullanıcaz**

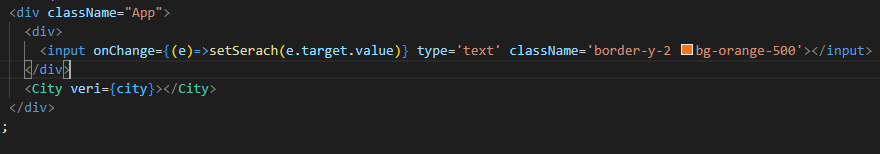
****

****

**Veri çekmek için: useEffect ve useState hooku oluşturuyoruz**

****

**City.jsx adında bir dosya indekxe import ettik oluşturduk bunu indexle bağdaştırcaz**

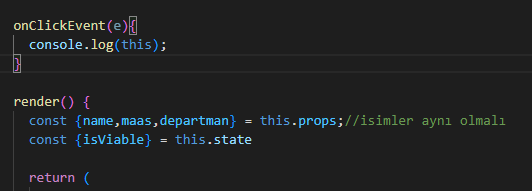
****

**Şimdi ise indeximizde bu City.jsx e city paremetresini gönderip geri dönüş beklicez**

****

**City.jsx te gelen veriyi işleyip return yapttık**

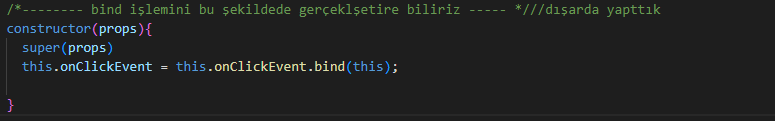
**Propslar companetler arası veri aktarımı**

****

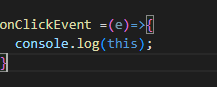
**Burda onclickEvent diye fonksiyon oluşturduk ama renderin dışında kaldığı için this diyerek  ana clasımıza erişemşyoruz bun için ;**

****

**Fonksiyonu çağırdığımız yere bind diyerek bulunduğu class objeleriyle birlikte gitmesini sağlıyoruz**

****

**Hiç aşşağıda tamamlamadan clasın içinde ve renderin dışında bir constructor tanımlayarakta yaparız**

****

**Yada firek olarak sadece arry function olarak tanımlarsak başka bir işlem yapmamamıza gerek kalmaz**

**Ama ek bilgi olarak funcitonların içine bir değer yazıcaksak ve bunu functiona gitmesini istiyorsak**

****

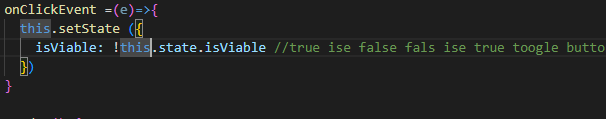
**İl olarak this parametresini ikinci olarak veriyi ekliyoruz**

  onClickEvent =(number,e)=>{

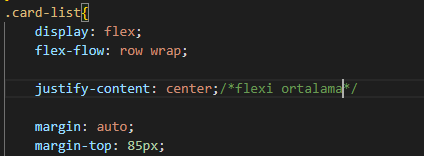
    console.log(number);

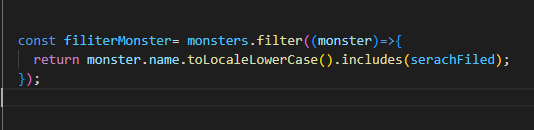
  }

**Fonksiyonun kendisinde ise ilk değişken yazılır sonra event kullanılacaksa yazılır**

****

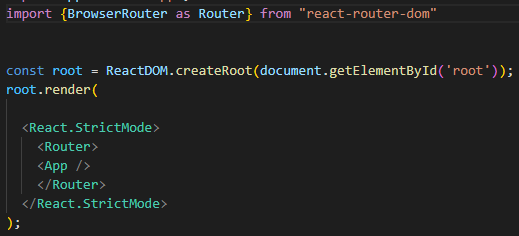
**setState ile erişim sağladıkve dğiştire bildik verileri state ler aracılıığıyıla geçişini ve haberleşmesini sağlıyacağız**

****

****

**Monster ın içinde 10 tane dizi objesi var bunuların nameylerini hepsini küçük har yaparak tek tek serchfiled da olan objeleri filitermosnstra atıyoruz**

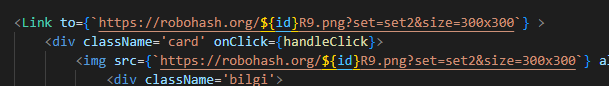
**Link işlemi için:**

****

**Reac-router-dom uu yüklüyoruz index.js sarmalıyoruz**

****

**Kullanıcağımız compannete Linki import ediyoruz**

****

**Bu şekilde kullanıyoruz**

****

**usRef kullanımı önce ekledik**

****

**oluşturduk**

****

**Ref diyip gereken yere ekliyoruz**

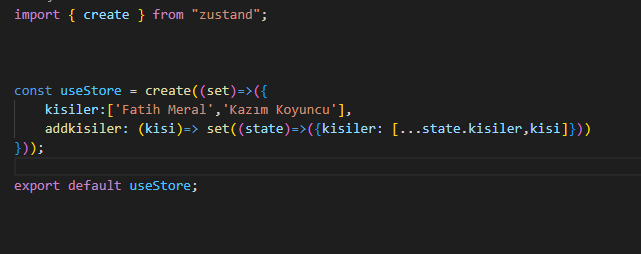
****

**Divin veya inputun özelliklerine erişim sağlaya biliyoruz**

**::this işleminin işe yaraamadığı yerlerde kullanıla bilinir::**

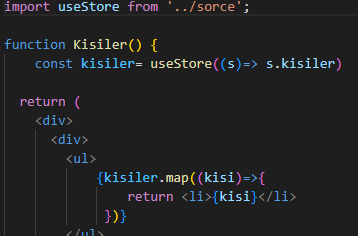
**Zustanda veri yazma işleminde sadece bu işlem işe yarar**

**Zustand:**

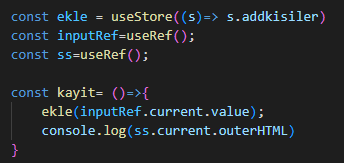
****

**…state.kisiler diyerek kişilerin devamına olarak ekleme işlemi yapttık**

**Bu şekilde sorce.js olarak oluşturduk**

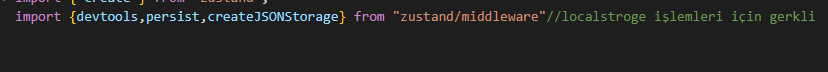
****

**Şu şekilde stateimizdeki kişiler verisini çektik**

****

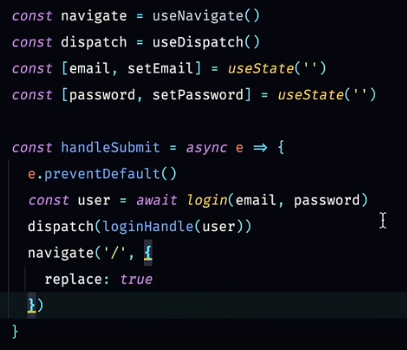
**Bu şekildede ekleme yapttık**

**Const içine bir kere atam yapılır let ise böyle bir kısıtlamaya tabi değildir**

****

****

**Bu şekil düzenledik ve artık localstorege kayıt oluyor sayfayı yenilesek bile hafızada kalıyor**

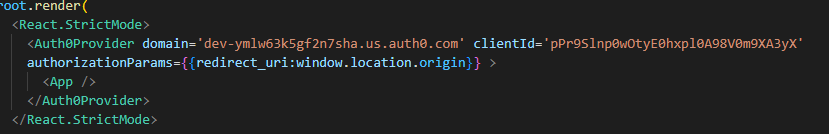
****

**Yönlendirme için**

**Manage Authencation auth0**

****

**İndex.js import ettik**

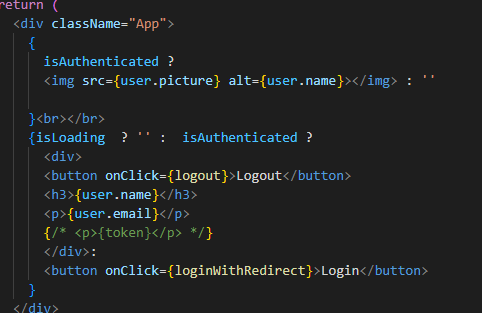
****

**Auth0prğviderla sarmallama yapttık domin client id ve hangi konumda açılcağını belirttik**

**App.js**

**ekledik**

****

****

loginWithRedirect

**login işlemi için**

logout

**çıkış işlemi için**

**userle belirli bi kaç ver, çekebiliyoruz**

isAuthenticated

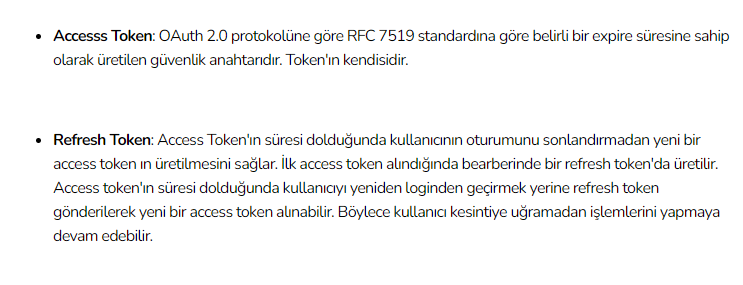
**Giriş olursa**

isLoading

**cevap bekleniyorkeenki gecikmee lütfen bekleyin gibi seyler yazıla bilinir**

**refresh token**

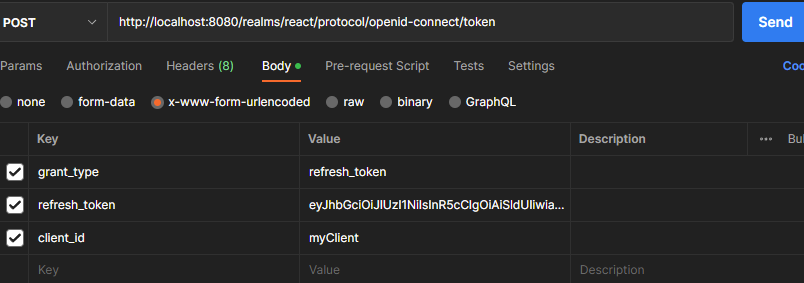
**Access token**

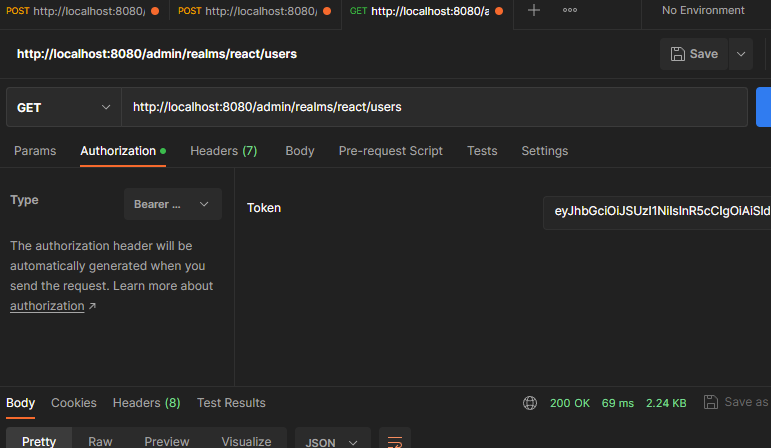
****

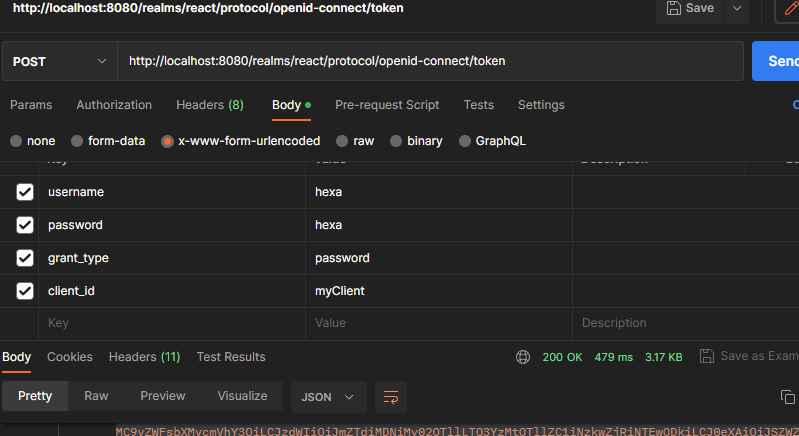
**Openid**

**Keycloack**

**Jwt (jasen web token)**

****

****

****