To Do App

1. React dosyamızı npx creat-react-app to-do-app diyerek kurduk.
2. Gereksiz dosyaları sildik
3. Sayfalar arası geçiş için npm i react-router-dom diyerek paketimizi indirdik
4. Sayfaları yerleştirebileceğimiz -pages dosayı -componenet dosyası -api buda api ler üzerinde veri çekmek istediğim için api doyası açıp içerisine api.js diye bir js dosyası açtık. Bu sayfada tüm api işlemlerini tamamlayıp diğer sayfalara kullanılabilir ve ulaşılabilir hale getireceğiz.
5. Pages altında introduce açacağız ve içine introduce.jsx dosyası oluşturacağız.

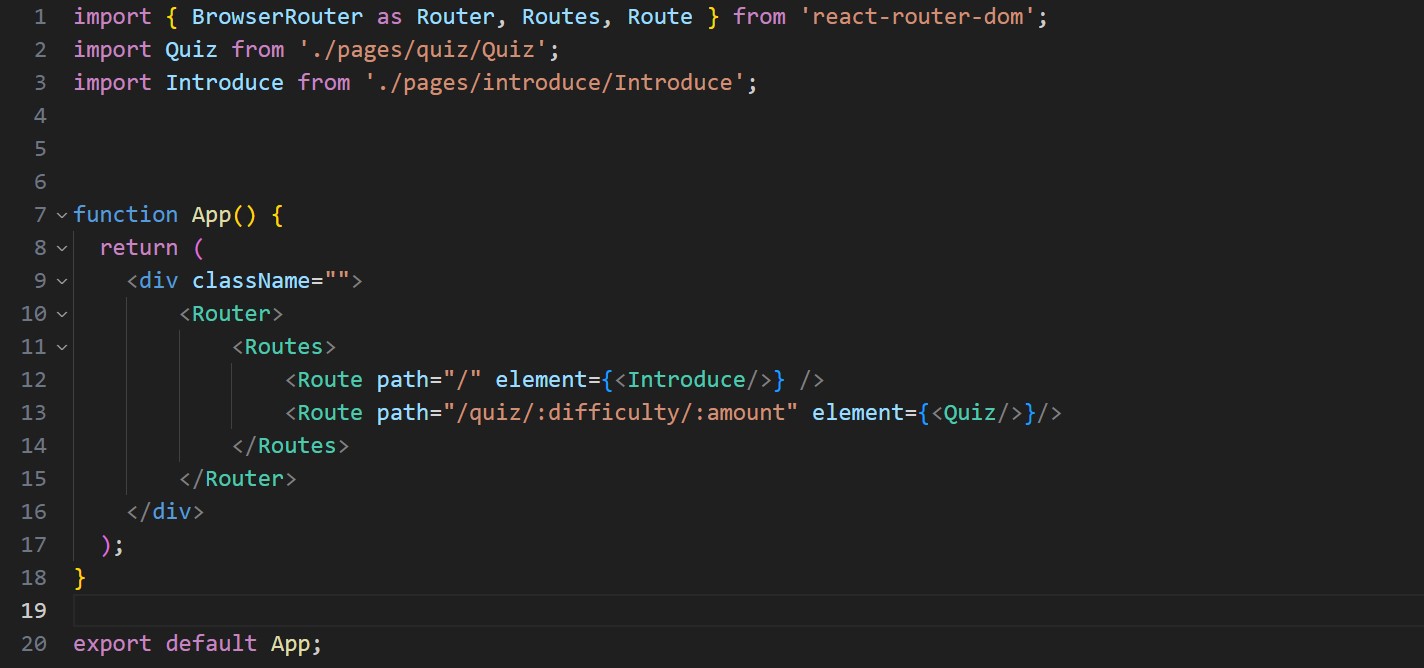
Ve altlarına introduce.css adına doya oluşturuyorum ve introduce.jsx içerisinden içerisinden ediyorum

1. Pages altına quiz dosyası açacağız ve içine quiz.jsx dosyası açıyoruz ve bir tane

Quiz.css adında bir dosya oluşturup quiz.jsx içerisinden import ediyoruz.

1. Şimdi app.js içerisinde react rooter doom ı kullanarak büütn yapıları bir birbirine bağlayacağız. Bağlama işlemini yapabilmek için

İmport {BrowserRouter as Router, Routes,Route} from ‘react-router-dom’ dememiz lazım.



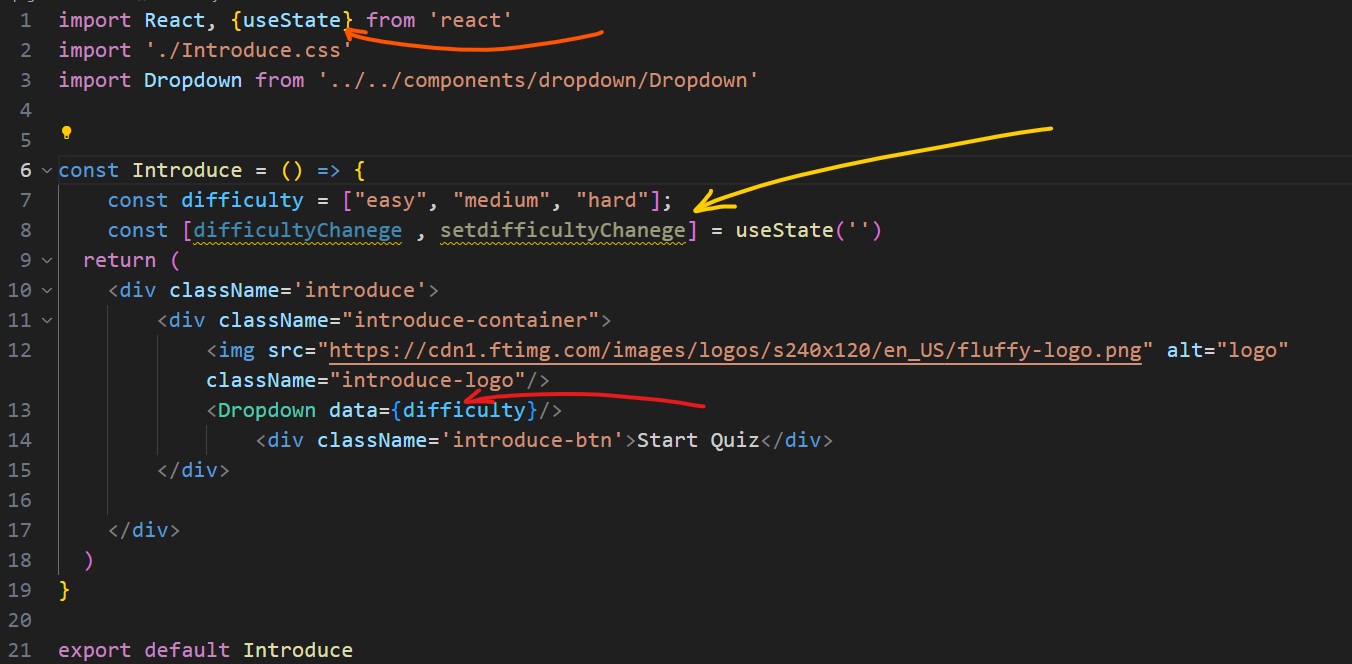
App.js içerisinde rooter map koyduk difficulty ve amount değerileri yazmama sebebmiz dışarıdan manuel olarak değerler gelecek olmasıdır. Burada değerler useNavigation ile gelir.

Aşağıda da kullanımını gösterildi zaten.

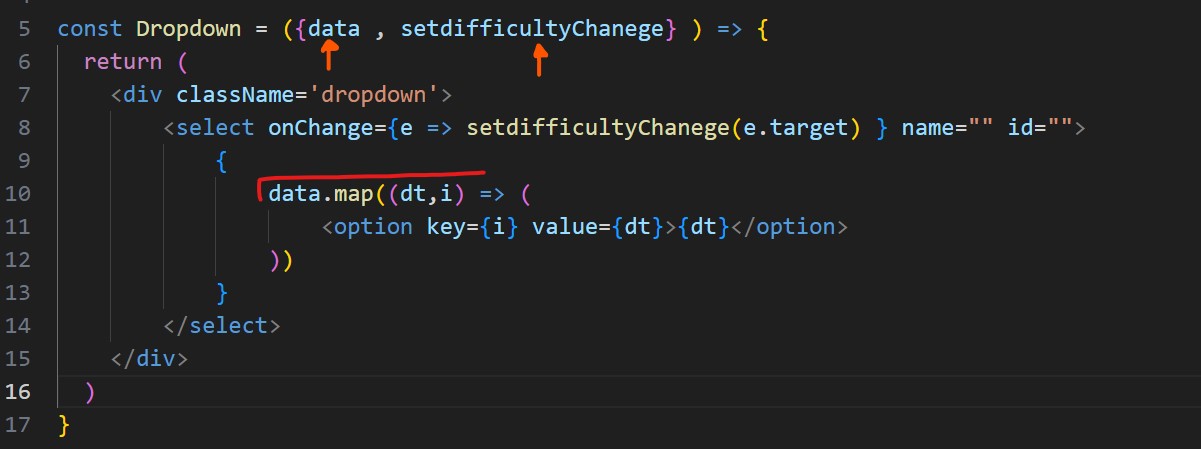
metin, ekran görüntüsü, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

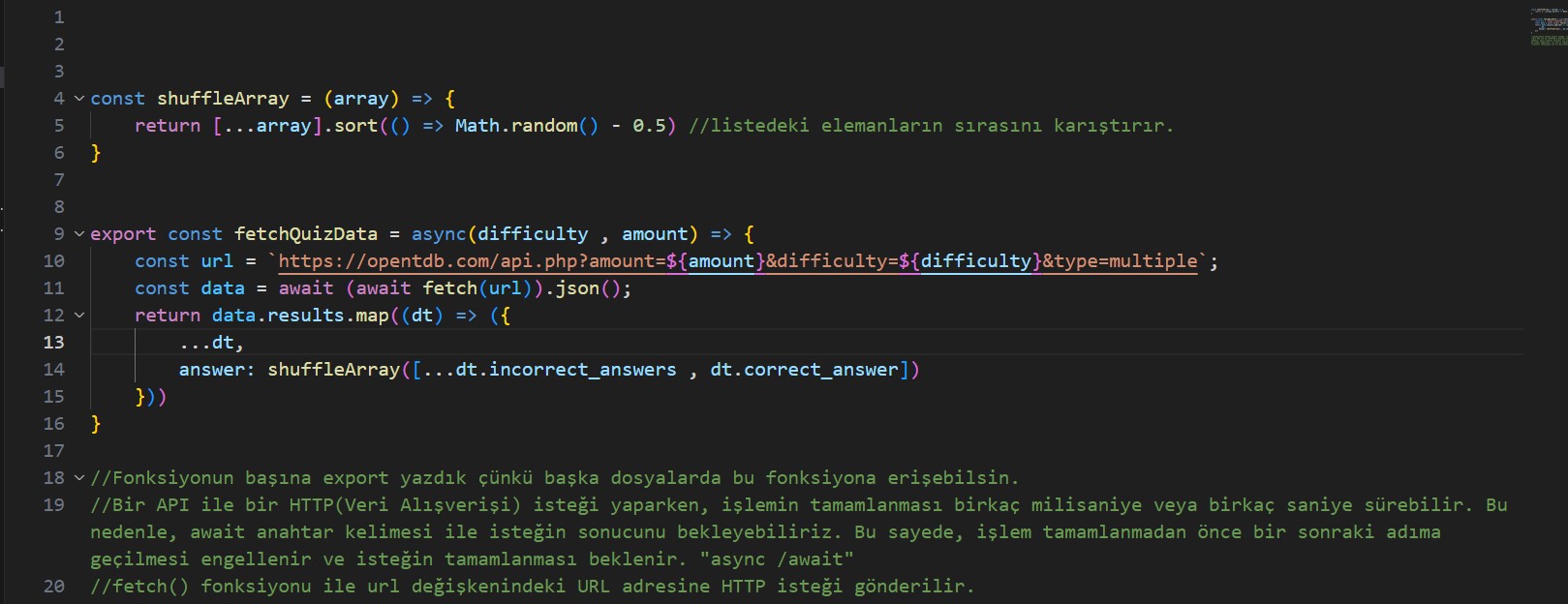
useNavigate bir değişkene atanarak kullanır.



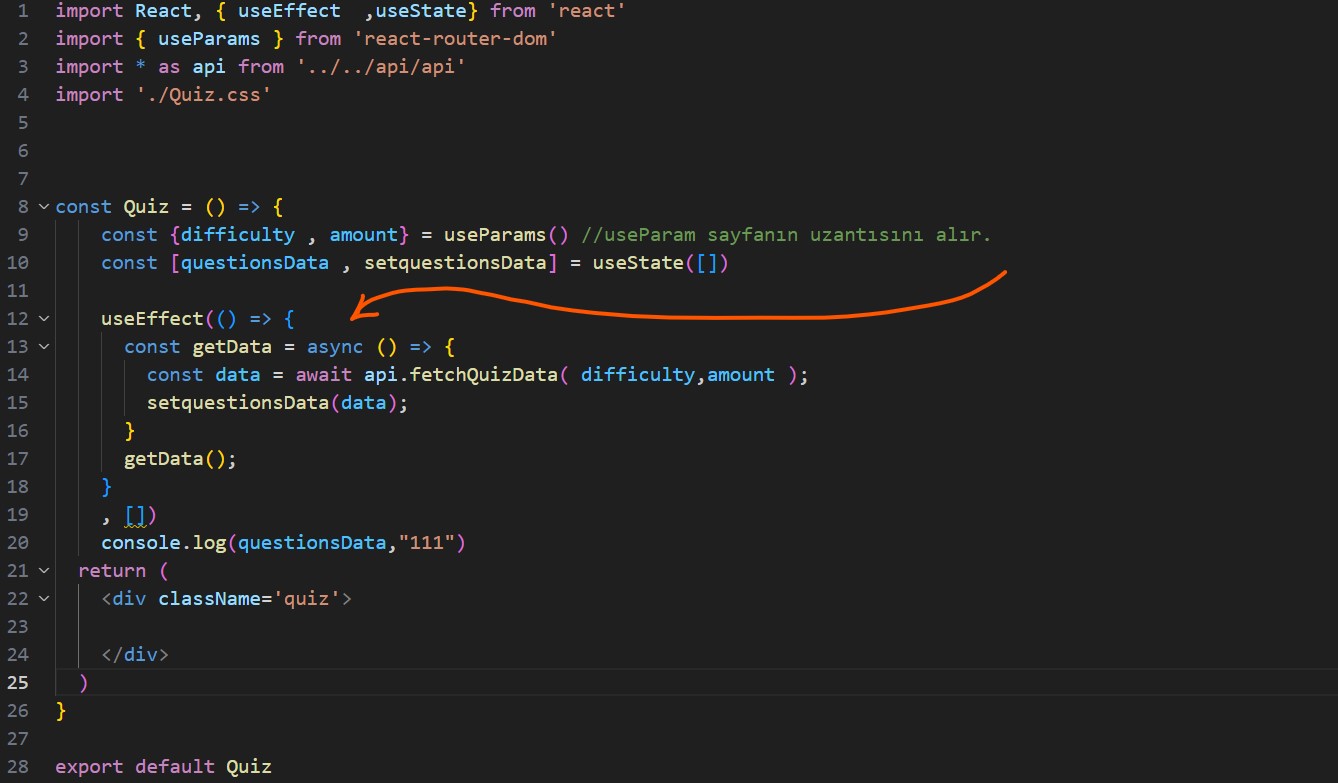
7. sırada oluşturmuş olduğum difficulty değişkenini kırmzı ok ile Dropdown dosyasında kullanılmak üzere değişken olarak atıyorum.



Gelen değerleri burada karşılıyorum. Diğer sayfada data içine yazdığım değerleri map komutu ile tek tek geziyorum option içerisine yazıyorum.



1. Şimdi de api.jsx içerisinde apilerimi çekeceğim. Export diyerek fonkisyonumu dışarıya açık bir fonksiyon yapıyorum.
2. Dışarıdan zorluk ve soru sayısını alıyoruz.
3. Api üzerinden almış olduğum linkide zoruluk ve soru sayılarını manuel olarak ekliyorum.
4. Fetch komutu ile link üzerinden veri alışverişi yapması komutunu verşyorum bu veriyi json formatına çevirmesini söylüyorum ve bu veri alışverişinden gelen bilgileri data adındaki değişkene atıyorum
5. Data üzerinden results (sonuçlar) a erişiyorum ve sonuçlar üzerinde tek tek geziniyorum. Gezerken bunların hepsini dt değişkenine atıyorum.
6. Bunların hepsini objenşn ilk parametresine kaydediyorum, daha sonra anserw adında bir key değeri oluşturupshuffleArray fonksiyonunu oluşturuyoruz.
7. Fonksiyonunun içerisine dt üzerinden yanlış ve doğru cevapları gönderiyoruz. Aslında her soruda 3 yanlış bir doğru olduğu için yanlış cevapların hepsini gönderme konusunda dikkat etmeliyiz bu yüzden de … kullanıyoruz.
8. Fonksiyonda arraydan gelen tüm doğru ve yanlış değerleri […array] yaparak başka bir dizi oluşturmuş oluyoruz
9. .sort(() => Math.random() – 0.5) ile dizi içindeki tüm değerlerin random dizilmesini sağlıyoruz.



Quiz.jsx içerisinde ise bu fonksiyonları ekliyoruz. İlk önce questionData adında useState hook fonksiyonu ile bir değişken oluşturuyoruz.

Daha sonra useEffect hook fonksiyonu ile sayfa yüklendiği zaman yapılacak fonksiyonlar söylendi bu fonksiyonlarda

getData adında bir fonksiyon oluştur data değişkenine api üzerinden.fetchQuizData fonksiyonuna difficulty ve amount değerlerini gönder. setquestionData araçılığı ile questionData değişkeninde bulunan boş dizinin içini sorularla doldurmasını istedim.