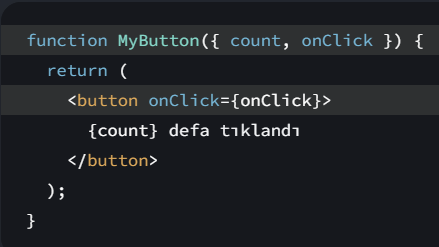
* <MyButton />’un büyük harf ile başladığına dikkat edin. Bu onun bir React bileşeni olduğunu belirtir.
* HTML elemanları küçük harf ile başlamak zorundayken, React bileşenleri her zaman büyük harfle başlamak zorundadır.
* Sınıf bileşenler yerine fonksiyonel bileşenler kullanın.
* Projelerde genellikle tek bir stil kullanılması tercih edilir. Bu nedenle, tırnak işareti seçimi genellikle ekip tarafından belirlenen kod stiline bağlıdır.
* export default bir componentin içinde olan bir çok fonksiyondan hangisinin varsayılan olarak çalıştırılacağını belirtir.
* Biçimlendirme sözdizimine JSX denir.
* React typescript cheat sheat <https://react-typescript-cheatsheet.netlify.app/docs/basic/setup>
* Döndürmek isteğiniz elemanları <div>...</div> veya boş <>...</> gibi ortak bir üst eleman içine koymanız gerekmektedir:
* Bileşeninizin bazı bilgileri ekran göstermek için “hatırlamasını” isteyeceksiniz. Örneğin, bir butona kaç defa tıklandığını takip etmek istiyorsunuz. Bunu yapmak için bileşeninize state ekleyin. (useState)
* const [count, setCount] = useState(0);
* useState’ten iki şey alacaksınız: mevcut state (count) ve state’i güncellemenizi sağlayan (setCount). Yaygın kullanış [something, setSomething] şeklindedir. Buton ilk defa görüntülendiğinde count değeri 0 olacaktır çünkü useState()’e 0 değerini ilettiniz.



* use ile başlayan fonksiyonlara Hook denir. useState, React tarafından sağlanan bir Hook’tur.
* Eğer useState’i bir koşul ya da döngü içinde kullanmak istiyorsanız, yeni bir bileşen oluşturun ve onun içine koyun.
* Her iki MyButton bileşeninin aynı count değerini göstermesi ve aynı anda güncellenmesi için state’i bireysel butonlardan “yukarı”, hepsini içeren en yakın bileşene taşımanız gerekmektedir.
* Yeni count değeri her bir butona **prop** olarak iletilir, böylelikle hepsi yeni değeri gösterir.





* State tanımlarken dikkat edilecek hususlar
  + Zaman içinde değişmeden mi duruyor? Eğer öyleyse, state değildir.
  + Prop’lar aracılığıyla üst bileşenden mi geliyor? Eğer öyleyse, state değildir.
  + Varolan state veya prop’lara dayalı olarak hesaplayabilir misiniz? Eğer öyleyse, kesinlikle state değildir!
* Eslint’in var olan ayarlarını kuralım
* Pretier kuralım
* useContext:
  + Bir temayı, dil ayarını veya kullanıcı bilgilerini tüm uygulama boyunca kolayca paylaşmak için kullanılabilir.
* useMemo:
  + React'te performans optimizasyonu yapmak için kullanılan bir Hook'tur. Bir bileşende hesaplanması maliyetli olan işlemlerin her render'da yeniden hesaplanmasını engelleyerek, yalnızca belirli bağımlılıkların değiştiği durumlarda bu hesaplamaların yapılmasını sağlar.
* useCallback:
  + useCallback, React'te performans optimizasyonu sağlamak amacıyla kullanılan bir Hook'tur. Özellikle bileşenlerin yeniden render edilmesi sırasında gereksiz fonksiyon yeniden oluşturulmalarını önlemek için kullanılır. useCallback, bir fonksiyonun bellekteki referansını hatırlar ve yalnızca belirli bağımlılıklar değiştiğinde bu fonksiyonu yeniden oluşturur.
* React developer tools’u kuralım