**2.**

Graphical user interface, application

Description automatically generated

Graphical user interface, application

Description automatically generated

Graphical user interface, text, application, website

Description automatically generated

**3.** [**https://www.typescriptlang.org/docs/handbook/react.html**](https://www.typescriptlang.org/docs/handbook/react.html)

**4.**

React.memo() is a [higher-order component (HOC)](https://reactjs.org/docs/higher-order-components.html), which is a fancy name for a component that takes a component as a prop and returns a component that prevents a component from re-rendering if the props (or values within it) have not changed.

**Wrapping up: The major differences between React.memo() and useMemo()**

From the example above, we can see the major differences between React.memo() and useMemo():

* React.memo() is a higher-order component that we can use to wrap components that we do not want to re-render unless props within them change

React.memo(), içindeki aksesuarlar değişmedikçe yeniden oluşturmak istemediğimiz bileşenleri sarmak için kullanabileceğimiz daha yüksek dereceli bir bileşendir.

* useMemo() is a React Hook that we can use to wrap functions within a component. We can use this to ensure that the values within that function are re-computed only when one of its dependencies change

useMemo(), bir bileşen içindeki işlevleri sarmak için kullanabileceğimiz bir Tepki Kancasıdır. Bunu, o fonksiyondaki değerlerin yalnızca bağımlılıklarından biri değiştiğinde yeniden hesaplanmasını sağlamak için kullanabiliriz.

* useReducer , geliştiricilere bileşenlerin state bir akış şeklinde yönetiliyorsa bu aşamada bir kolaylık sağlar **[useState](https://medium.com/frontend-development-with-js/hooks-usestate-nedir-9b7ae42a6895)**yerine bu Hook’tan faydalanabilirsiniz. Örneğin aşağıdaki örnekte Ekranımızda bir counter değeri var. Buna etki eden UI Bileşenleri var. En basit yöntem bunu useState üzerinden kullanarak 1 arttırmak, 1 azaltmak veya 0 set etmek olarak düşünülebilir.

A picture containing background pattern

Description automatically generated

Fakat useReducer bu ve daha kompleks State geçişleri için geliştiricileri bunu aynen Redux mekanizmasındaki benzer şekilde dispatch mantığı ile yönetmenize imkan sağlar.

7.

Text

Description automatically generated

10.

#### **Eklenmesi**

Bir bileşenin oluşumundan ve DOM’a eklenmesine kadar geçen süreç içerisinde çağrılan metotlar, sırasıyla aşağıdaki gibi belirlenmiştir:

* [**constructor()**](https://tr.reactjs.org/docs/react-component.html#constructor)
* [static getDerivedStateFromProps()](https://tr.reactjs.org/docs/react-component.html#static-getderivedstatefromprops)
* [**render()**](https://tr.reactjs.org/docs/react-component.html#render)
* [**componentDidMount()**](https://tr.reactjs.org/docs/react-component.html#componentdidmount)

#### **Güncellenmesi**

Bileşenin güncellemesi, kendi props’u veya state’i üzerindeki değişikliklerden oluşabilir. Bir bileşen tekrar render edildiğinde çağrılan metotlar sırasıyla aşağıdaki gibidir:

* [static getDerivedStateFromProps()](https://tr.reactjs.org/docs/react-component.html#static-getderivedstatefromprops)
* [shouldComponentUpdate()](https://tr.reactjs.org/docs/react-component.html#shouldcomponentupdate)
* [**render()**](https://tr.reactjs.org/docs/react-component.html#render)
* [getSnapshotBeforeUpdate()](https://tr.reactjs.org/docs/react-component.html#getsnapshotbeforeupdate)
* [**componentDidUpdate()**](https://tr.reactjs.org/docs/react-component.html#componentdidupdate)

#### **Çıkarılması**

Bir bileşen, DOM’dan çıkarıldığında bu metot çalışır:

* [**componentWillUnmount()**](https://tr.reactjs.org/docs/react-component.html#componentwillunmount)

#### **Hatanın Yakalanması**

Aşağıdaki metotlar; render esnasında, yaşam döngüsü metodunda veya herhangi bir alt bileşenin constructor’ında bir hata oluştuğunda çağrılmaktadır:

* [static getDerivedStateFromError()](https://tr.reactjs.org/docs/react-component.html#static-getderivedstatefromerror)
* [componentDidCatch()](https://tr.reactjs.org/docs/react-component.html#componentdidcatch)