

# What's **useRef** hook?



## **useRef**

When working with functional components, you often need to compare the current state or props with the previous ones. A lesser-known but super useful trick is to use **useRef** to store the previous state or props.

**useRef** is a hook in React that provides a way to create a mutable reference object. This reference object persists across renders and can be used to store values that don't trigger re-renders when changed. Here's a brief overview:



[/in/mulukennm](https://www.linkedin.com/in/mulukennm)

# Description of useRef

- **Purpose:** useRef is primarily used to access and interact with DOM elements directly or to keep a mutable value that doesn't cause a re-render when updated.
- **Return Value:** It returns a mutable ref object with a `.current` property that can hold any value.

## Why It's Useful:

- **Efficient State Comparison:** This method allows for efficient comparisons between previous and current states or props without the overhead of extra state management.
- **Avoids Unnecessary Renders:** Since useRef does not trigger re-renders, it's an optimal way to keep track of values between renders.
- **Simplifies Logic:** It can simplify the logic in your components by avoiding complex state management patterns.



# Here's how you can use it:

```
import { useEffect, useRef } from 'react';

const MyComponent = ({ someProp }) => {
  const prevPropRef = useRef();

  useEffect(() => {
    // Access previous prop
    const prevProp = prevPropRef.current;

    // Do something with prevProp and someProp

    // Update ref to current prop
    prevPropRef.current = someProp;
  }, [someProp]);

  return <div>{/* Your component JSX */}</div>;
};
```

## Explanation of the above code

- **useRef**: Unlike state, changing a ref does not cause a re-render. This makes it perfect for storing values that persist between renders but do not directly affect the UI.
- **useEffect**: Inside the useEffect hook, you can access the previous value of the prop before updating the ref to the current prop. This runs every time someProp changes.
- **prevPropRef.current**: This holds the previous value of someProp and gets updated after each render cycle.

