**Use Memo:**

Like other hooks useMemo is a hook that is used to memorize things (caching).  
 The main purpose of this hook is to add optimization to the existing react code.

We can use the “useMemo” Hook whenever there is a heavy computation which is making the overall user experience slow then we can use this to stop unnecessary recomputing.

One more use case:  
 We can keep the referential variables like arrays objects etc. in useMemo because on every render a new reference to the variable is created.

Now the question arises why don’t we useMemo everywhere since it will eventually be fast.

1 Because it needs some extra memory to store the precomputed values.  
2 on each render useMemo will be called.

Examples:  
1. We can do memoization for heavy functions.

2. We can do memoization for referential childs.

3. We can do memoization for even a react child

Write some examples.

**useRef:**

It’s just like state the value of refs persist between re renders but the main difference is refs do not cause re renders.

It returns an object with only one property “current”

The value will persist between re renders but change in ref value will not cause a re render.

Main use case for ref’s is to reference HTML elements inside the DOM.

We can get access to any element in which ref is passed.

Refs can be used to store the value of previous state, without causing a re render

Generally we do on HTML elements

**forwardRefs:**

Till now we saw that we assigned refs to pure html components only.

but in case of react we will be having components where we may want to use refs

Forward refs are used to pass refs to components so that the inner html can be accessed through that refs.

**useImperativeHandle:**

This hook is not used much.

This is used in combination with forward ref.

This hook is used to define custom definition on refs for parent component passing them

**React Portals**

It is a great way to create modals in react.

Event bubbling is always captured by the parent from where we laid the portal.