**React**

**Day 43rd**

**React Refs**

Refs provide a way to access DOM nodes or react elements created in the render method. In the typical react dataflow, props are only way that parent components interact with their children. To modify a child, you re-render it with new props. However, there are a few cases where you need to imperatively modify a child outside of the typical dataflow. The child to be modified could be an instance of a React component, or it could be a DOM element. For both of these cases, React provides an escape hatch.

Refs are a function provided by react to access the DOM element and the react element that you might have created on your own. They are used in cases where we want to change the value of a child component without making use of props and all.

**When to use refs**

There are a few good use cases for refs:

* Managing focus, text selection, or media playback.
* Triggering imperative animations
* Integrating with third party DOM libraries.

Avoid using refs for anything that can be done declaratively.

React provides a built-in function called createRef() that allows us to create a reference (ref) to access a DOM element or a React component directly. We attach this ref to a JSX element using the ref prop, which is a special built-in prop available on all React-supported HTML elements. The ref prop behaves like an attribute and allows us to reference the underlying DOM node or component instance.

**Don’t overuse refs**

Your first inclination may be to use refs to make things happen in your app. If this is the case, take a moment and think more critically about where state should be owned in the component hierarchy. Often, it becomes clear that the proper place to own that state is at higher level in the hierarchy.

**How to create ref??**

React provide us createRef() function. This function allows us to create ref object in ReactJS. It is available in CBC and FBC. You need to attach this to either HTML or JSX element with ref prop (available in all tag) after creating.