REACT

ref

React supports a special attribute that you can attach to any component. The ref attribute takes a callback function, and the callback will be executed immediately after the component is mounted or unmounted.
When the ref attribute is used on an HTML element, the ref callback receives the underlying DOM element as its argument.

In the typical React dataflow, props are the 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
There are a few good use cases for refs:
•"Ö æ v–ær fö7W2Â FPxt selection, or media playback.
••@riggering imperative animations.
•"–çFPgrating with third-party DOM libraries.

React will call the ref callback with the DOM element when the component mounts, and call it with null when it unmounts.
Using the ref callback just to set a property on the class is a common pattern for accessing DOM elements.

•F†R &VfW'&VB pay is to set the property in the ref callback like in the above input}.
•
•v†Vâ F†R &Vb GG&– ute is used on a custom component declared as a class, the refcallback receives the mounted instance of the component as its argument.
______***___***___***___***___***___***