React uses a **reconciliation algorithm** to determine how to update the **DOM** when state or props change in a component. The goal is to make the process of updating the UI as efficient as possible, minimizing unnecessary re-renders and updates to the actual DOM.

import React, { useState } from "react";

function App() {

  const [count, setCount] = useState(0);

  const increment = () => setCount(count + 1);

  return (

    <div>

      <h1>Counter: {count}</h1>

      <button onClick={increment}>Increment</button>

    </div>

  );

}

export default App;

 Initially, React creates a Virtual DOM tree representing the UI structure (<div>, <h1>, <button>).

 When the state (count) is updated by clicking the button, React creates a new Virtual DOM.

 React compares the new Virtual DOM with the previous one and identifies that the only thing that changed was the value inside the <h1>.

 React efficiently updates the actual DOM to reflect this change in the UI without re-rendering the entire page.