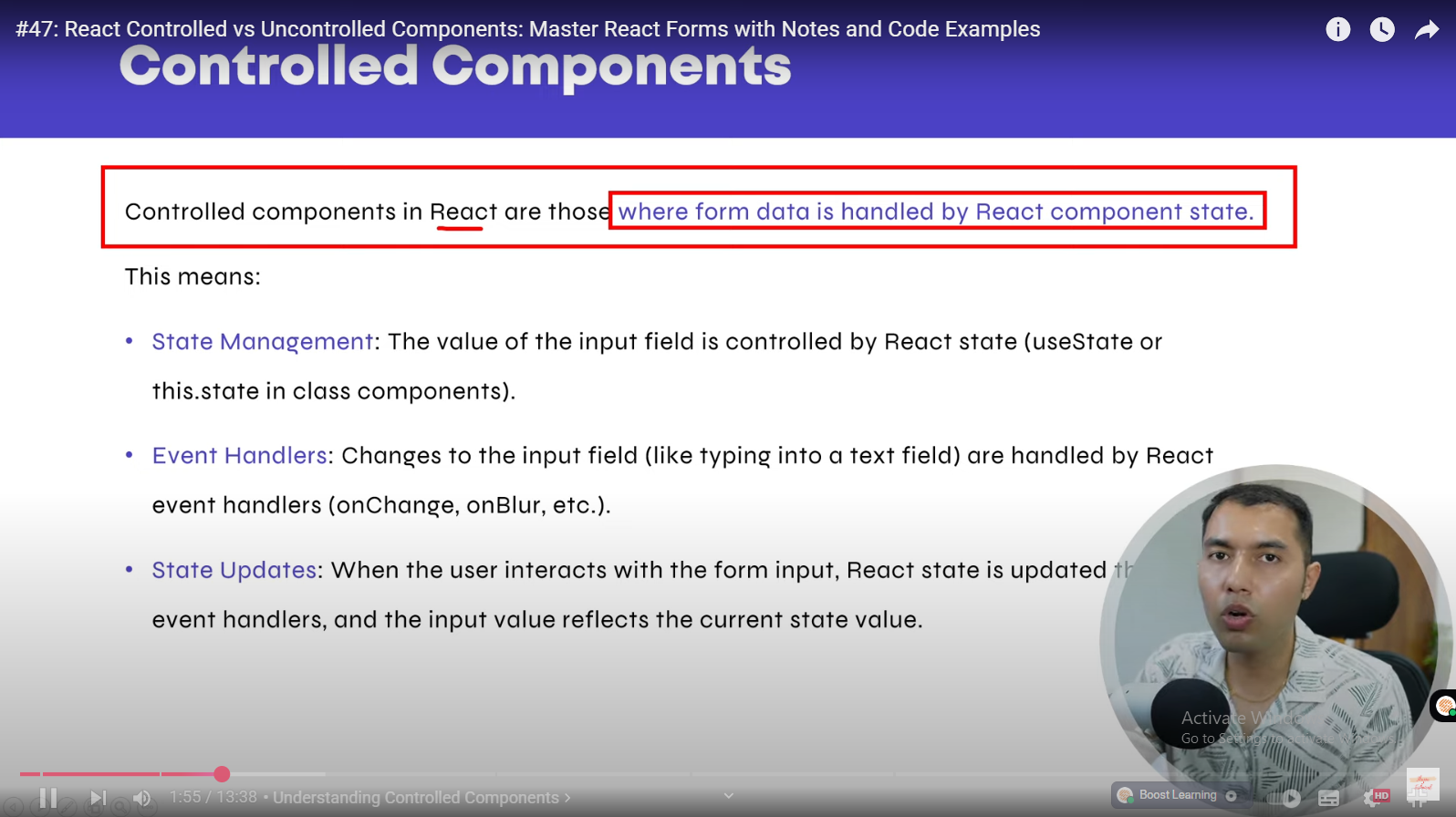
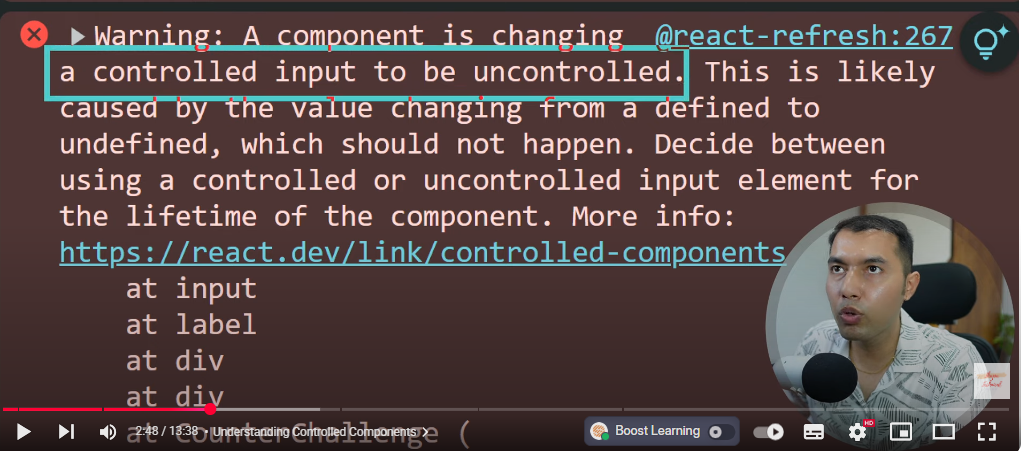
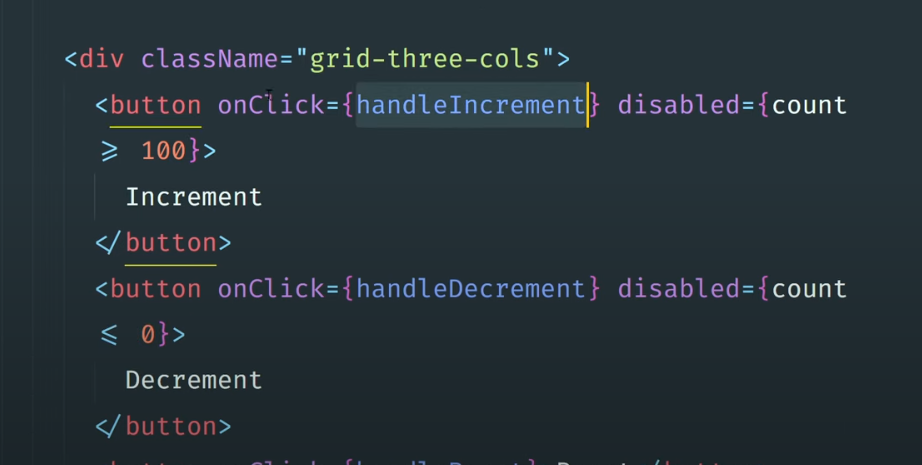
1. React Fundamentals  
   What is React? Key features and advantages  
   Real DOM vs Virtual DOM  
   JSX: Syntax & benefits  
   Components: Functional vs Class components  
   Props and State: Differences and usage  
   Component composition and reusability  
     
   2. Component Lifecycle & Hooks  
   Traditional lifecycle methods (for legacy code):  
   componentDidMount, componentDidUpdate, componentWillUnmount, etc.  
   Modern lifecycle with Hooks:  
   useState  
   useEffect (side effects, cleanup, dependency array)  
   useContext  
   useReducer  
   useRef  
   Custom Hooks  
   Lifting state up  
     
   3. Advanced React Concepts  
   Controlled vs Uncontrolled components

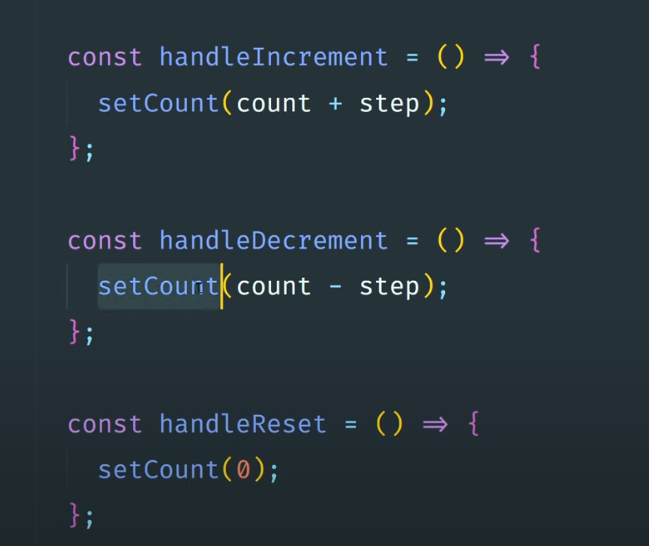






Q k hamne value ko comment out kia tou warning me wo bata raha ha aapne aik controlled component ko jo k react k useState hook se ham control kr rahe the uey uncontrolled banadia yani react ab usey manage nhi krega balke wo automaticaaly manage hoga by DOM.

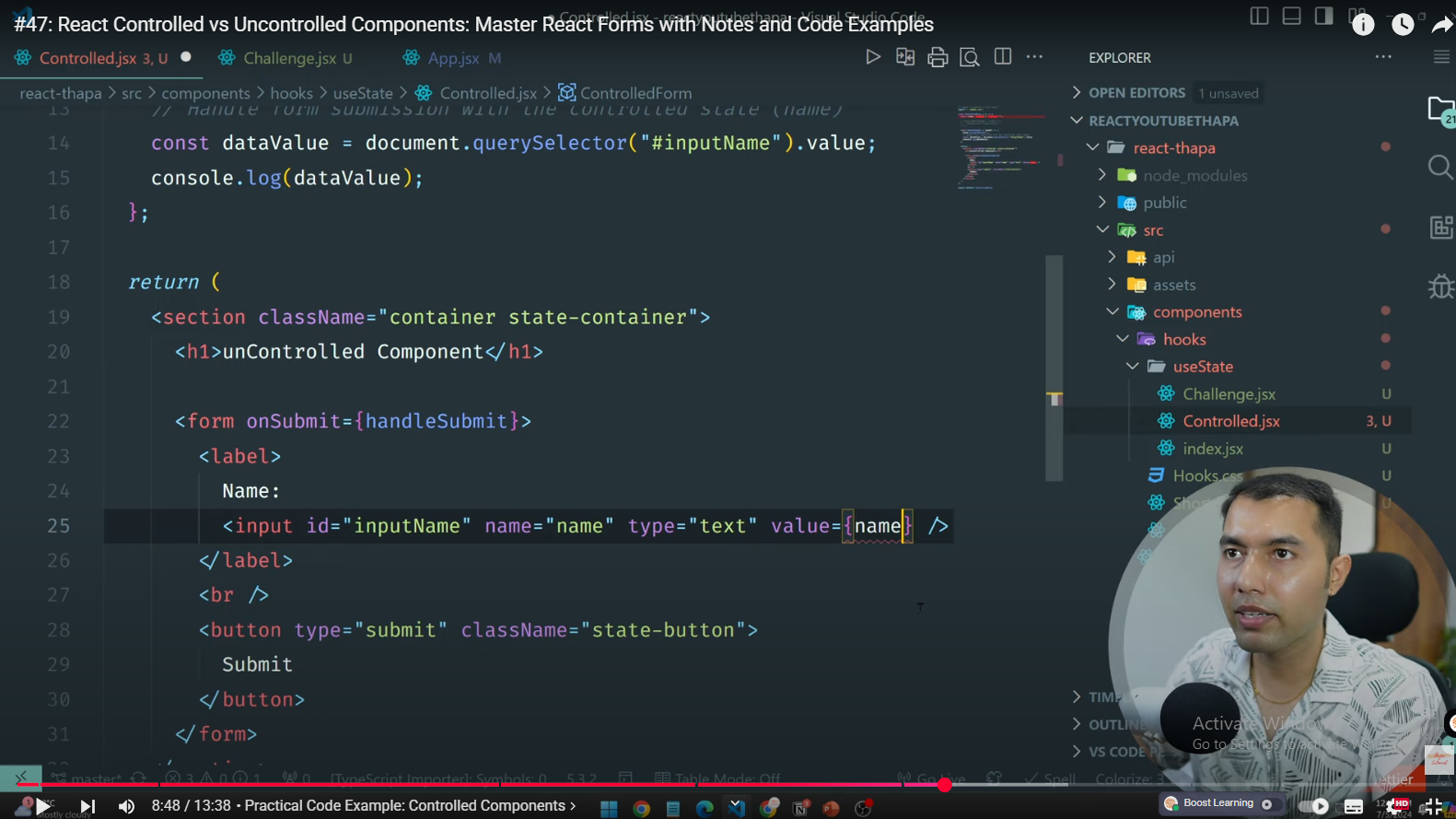


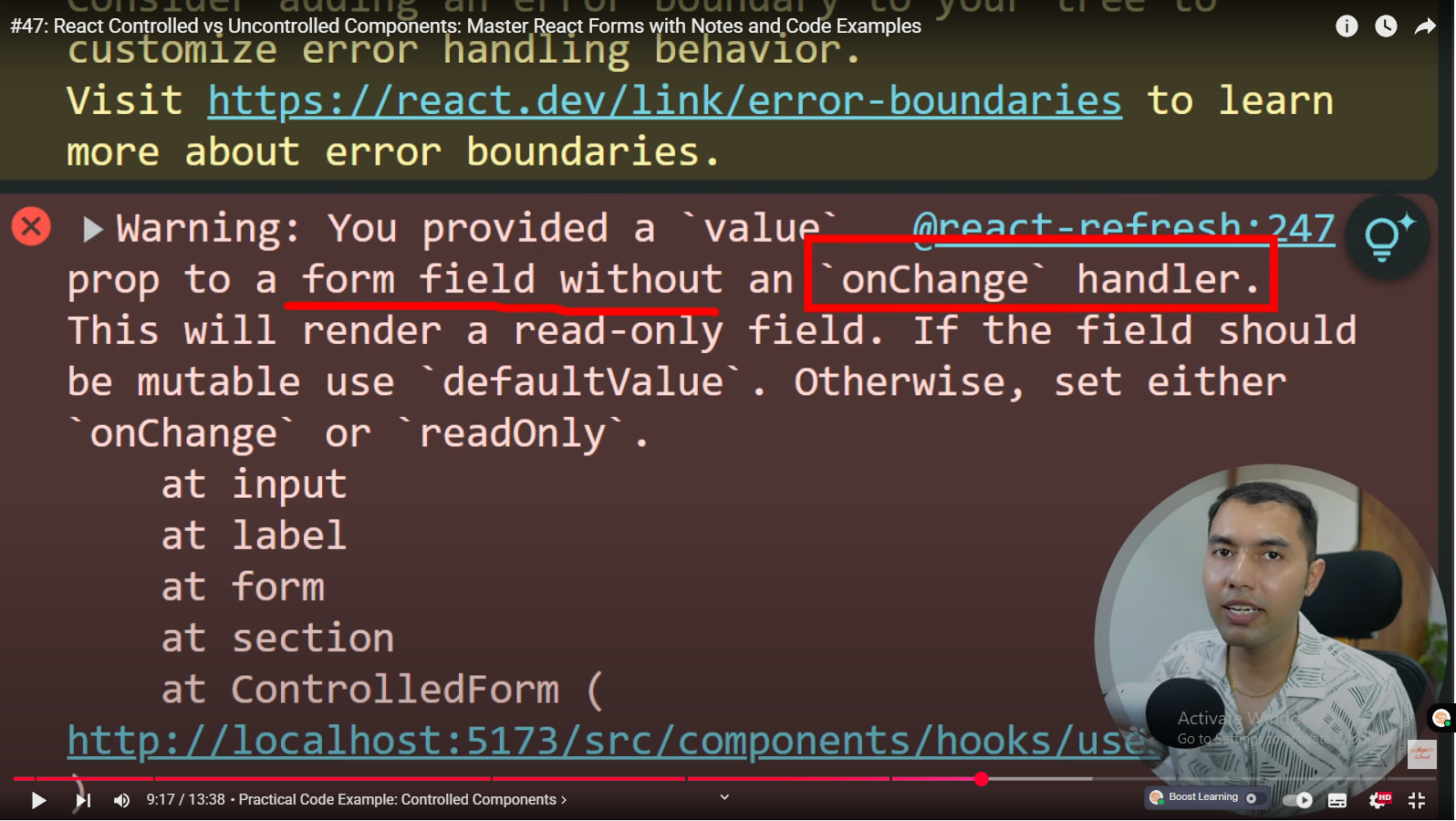


Value aap update kr rahe ho using vent handler , ye onclick ,onchnage pe aapne khud handler banaya wa ha react me jo value update kr raha ha increase ya decrease so this is controlled component

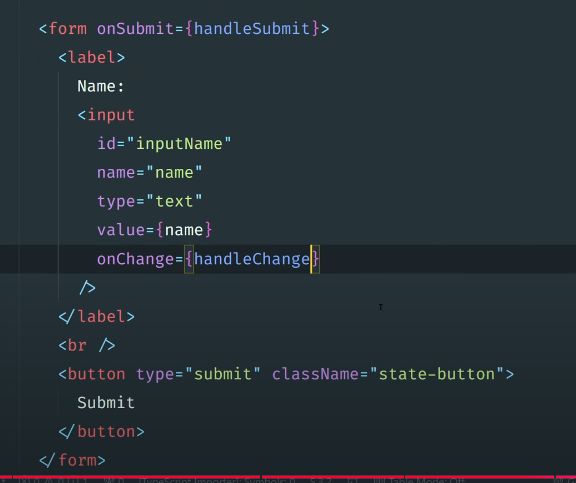
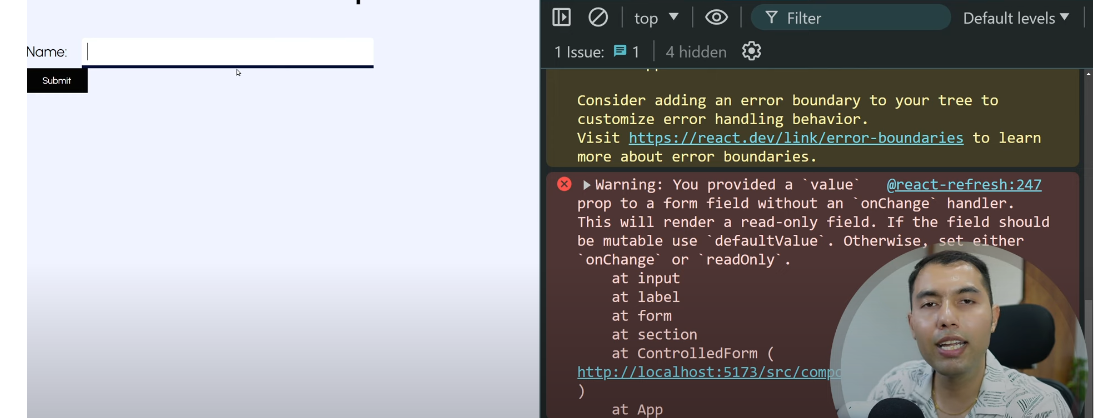
Note: form me name attribute use krna must ha aur is name attribute ki value aur state variable ka name same hona chahye

e.g demo:

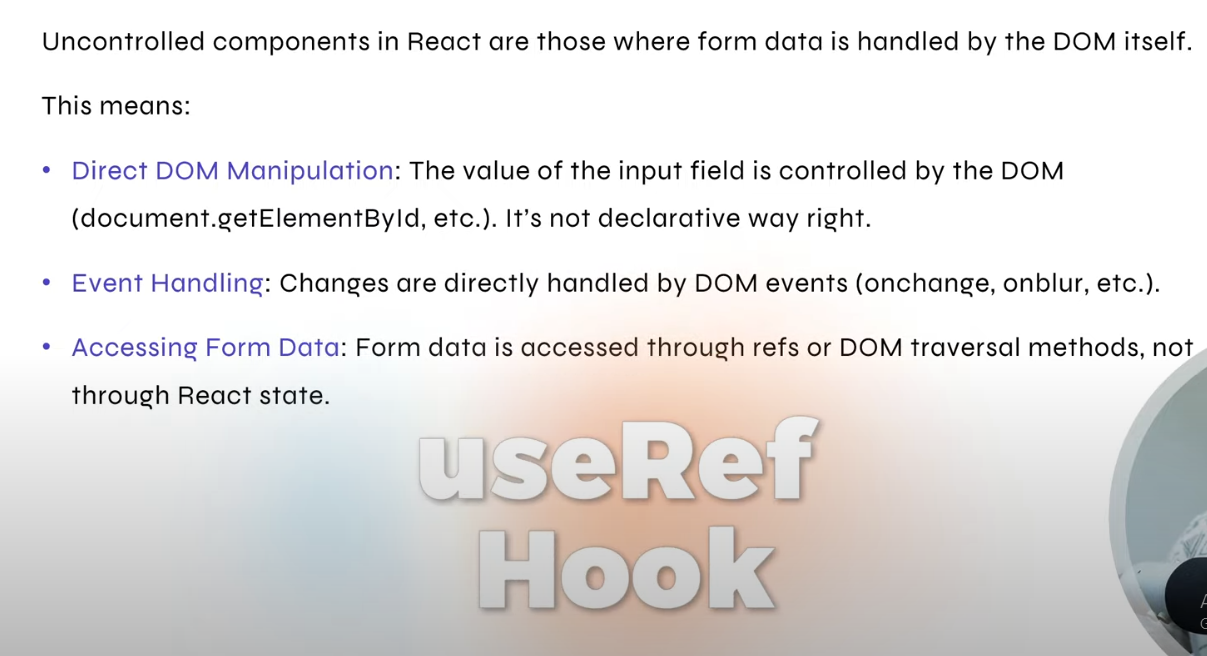


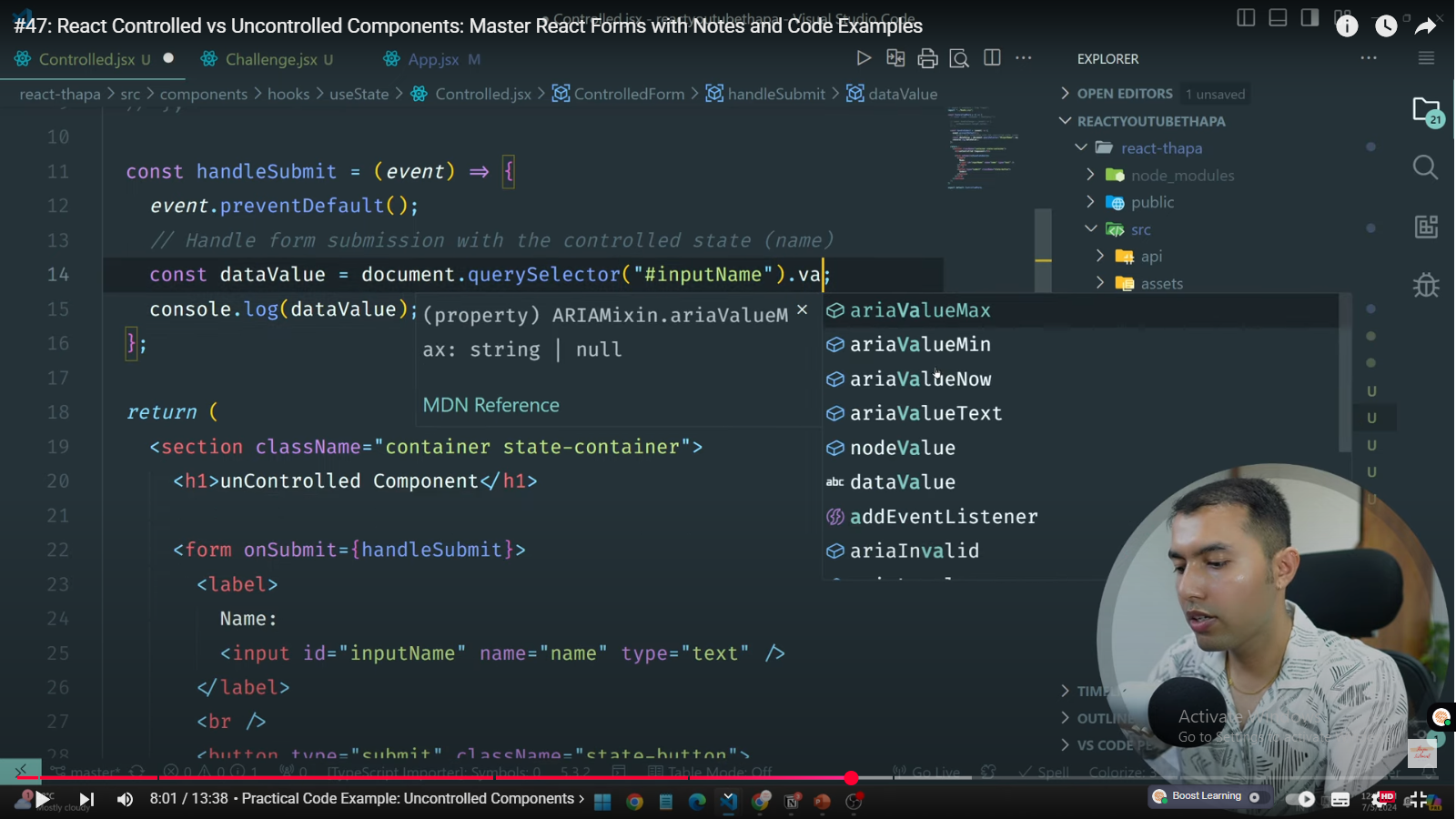


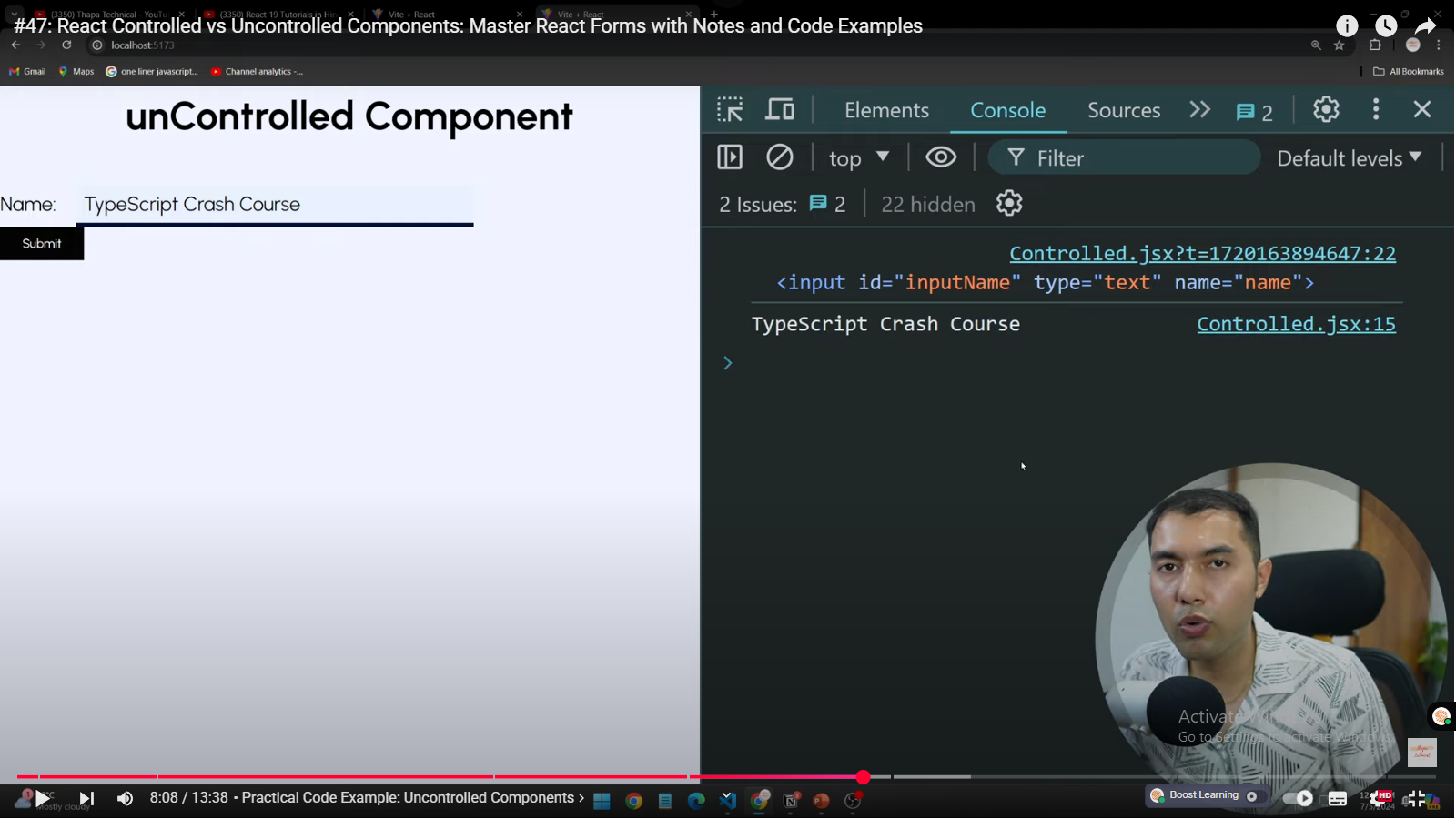
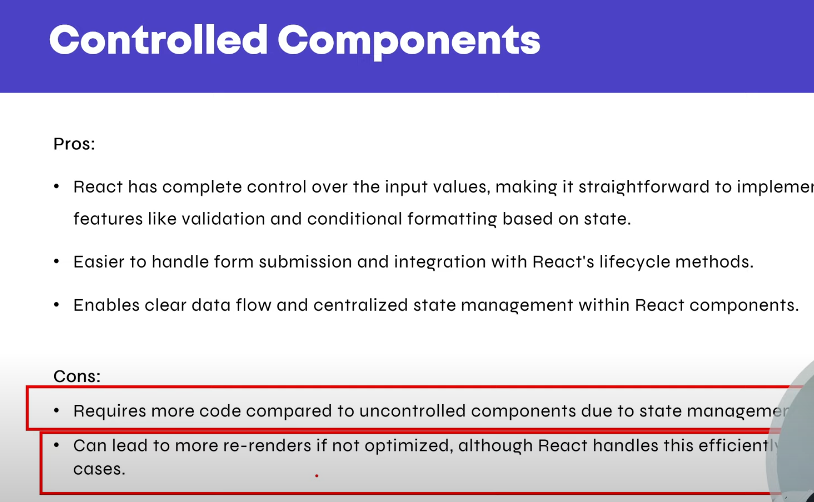
Abhi ye kia ha k aik state banai name ki aur input field me value = {name } but onchnage handler nhi lagaya tou code khud warning de raha ha , user input field me kuch likh hi nhi paraha

Uncontrolled components:





Prop drilling and Context API  
Error boundaries  
Higher-Order Components (HOC)  
Render props pattern  
React.memo, useMemo, useCallback for performance  
Code splitting: React.lazy, Suspense  
React Portals  
  
4. Forms & Events  
Handling forms and input validation  
Controlled form elements  
Event handling in React  
  
5. Routing & Navigation  
React Router basics:  
Routes, Link, Navigate, useNavigate  
Nested routes and dynamic routing  
Redirects and 404 pages  
  
6. State Management  
Local state vs global state  
Context API for state sharing  
Redux basics: Store, Actions, Reducers, Middleware  
Alternatives: Zustand, Recoil, Jotai, etc.  
  
7. Performance Optimization  
Avoiding unnecessary re-renders  
List virtualization/windowing (e.g., react-window)  
Lazy loading and bundle optimization  
Memoization strategies  
  
8. Testing  
Unit testing with Jest  
Component testing with React Testing Library  
Mocking and snapshot tests  
  
9. Ecosystem & Tooling  
Create React App, Vite, Next.js basics  
NPM/Yarn package management  
Webpack and Babel basics  
ESLint, Prettier, and code quality tools  
  
10. Real-World Scenarios  
Authentication and protected routes  
API calls (fetch, axios) and handling async data  
Error handling and fallback UI  
Responsive design and CSS-in-JS (styled-components, emotion)  
  
11. Miscellaneous  
PropTypes and TypeScript integration  
Accessibility (a11y) in React apps  
Best practices for folder structure and large codebases  
Deployment basics (Netlify, Vercel, etc.)