

[Github URL](#)

[Github URL for Vercel](#)

[Vercel URL](#)

Video: W13-P1: Demo prop drilling

=> Chrome, remove 4th person

The screenshot shows a browser window with a React application titled "Prop Drilling". The application displays a list of three names: John, Peter, and Susan, each with a "Remove" button below it. The entire list is enclosed in a red box.

On the right side of the screen is the React DevTools component tree and state inspection tool. The component tree shows the following structure:

- <ClientPageRoot>
- <PropDrilling_02 Memo>
- <Tutorials_xx_Wrapper>
- <List Memo>
- <SinglePerson key="1" Memo>
- <SinglePerson key="2" Memo>
- <SinglePerson key="3" Memo>
- <Next.MetadataOutlet Server>
- <_next_outlet_boundary>
- Suspense name="Next.MetadataOutU..." SegmentBoundaryTriggerNode SegmentViewStateNode key="/tutorials/cont..."
- BoundaryTriggerNode key="/tutorials/cont..."

The state inspection tool shows the following data under "props":

```
props
params: bound Promise
searchParams: bound Promise
new entry: ""
```

Under "hooks", the "useState" hook is shown with the following state:

```
1 State: [[-, -, -]]
  × 0: {id: 1, name: "john"}
  × 1: {id: 2, name: "peter"}
  × 2: {id: 3, name: "susan"} new entry
```

The "source" section points to "page.jsx:11".

At the bottom of the browser window, there is a toolbar with various icons and a status bar showing the date and time.

=> relevant code

```
File Edit Selection View Go Run ... 1141_2n_next_apple.02 EXPLORER OPEN EDITORS > 1141_2N_NEXT_APPLE_02 > .next > db > node_modules > public > src > app > (root) > tutorials > _assets > _data > JS data.js > _wrapper > Tutorials_xxjsx > context \_propdrillin... > page.jsx > favicon.ico > # globals.css > JS layout.js > JS page.js > exams > assets > components > midterm > quiz1 > tutorials > CODEX: CODEX > OUTLINE > TIMELINE > pagejsx U > layout.js U > Navbar_02.jsx U > ... > (root) > tutorials > context > 1_propdrilling_02 > page.jsx > PropDrilling_02 cases) 10 11 const PropDrilling_02 = () => { 12   const [people, setPeople] = useState(data) 13   console.log('data', data) 14 15   const removePerson = (id) => { 16     setPeople((people) => { 17       return people.filter((person) => person.id !== id) 18     }) 19   } 20 21   return ( 22     <Wrapper> 23       <div className='container'> 24         <h3>prop drilling</h3> 25         <List people={people} removePerson={removePerson} /> 26       </div> 27     </Wrapper> 28   ) 29 30 31   const List = ({ people, removePerson }) => { 32     return ( 33       <> 34         {people.map((person) => { 35           return ( 36             <SinglePerson key={person.id} {...person} removePerson={removePerson} /> 37           ) 38         }) 39       </> 40     ) 41   ) 42 } 43 44 } 45 46 47 const SinglePerson = ({ id, name, removePerson }) => { 48   return ( 49     <div className='item'> 50       <h4>{name}</h4> 51       <button className='btn' onClick={() => removePerson(id)}> 52         remove 53       </button> 54     </div> 55   ) 56 } 57 58 export default PropDrilling_02 59
```

fa83134 apple550678

2025-12-10 20:37:05 +0800

Video: W13-P1: Demo prop drilling

Video: W13-P2: Context API demo for W13-P1

=> Chrome, show PeopleContext.Provider by remove last person

The screenshot shows a browser window with the URL `localhost:3000/tutorials/context/2_context_people_02`. The page displays a list of names: John, Peter, and Susan, each with a "Remove" button. The "Context API Demo" title is highlighted with a red box. To the right, the React component tree in the DevTools Components tab is shown, also with a red box highlighting the `Context.Provider` component. The props of this provider are expanded, showing an array of people: `[{id: 1, name: "john"}, {id: 2, name: "peter"}, {id: 3, name: "susan"}]`. The "removePerson" prop is defined as `t1() {}`.

```
Context API Demo
John
Peter
Susan
Remove
Remove
Remove

Context.Provider
props
  children: <ForwardRef />
  value: [people: Array(3), removePerson: <...>]
  people: [{}, {}, {}]
    0: {id: 1, name: "john"}
    1: {id: 2, name: "peter"}
    2: {id: 3, name: "susan"}
```

=> relevant code

```
src > app > (root) > tutorials > context > 2_context_people_02 > _con
src > app > (root) > tutorials > context > 2_context_people_02 > page.jsx > PropDrilling_02
import { useState, createContext, useContext } from 'react'
import { data } from '../../../../../_assets/_data/data'

const PeopleContext = createContext()

export const PeopleContextProvider_02 = ({ children }) => {
  const [people, setPeople] = useState(data)
  console.log('data', data)

  const removePerson = (id) => {
    setPeople((people) => {
      return people.filter((person) => person.id !== id)
    })
  }

  return (
    <PeopleContext.Provider value={{ people, removePerson }}>
      {children}
    </PeopleContext.Provider>
  )
}

export const usePeopleContext_02 = () => {
  return useContext(PeopleContext)
}

import { usePeopleContext_02, PeopleContextProvider_02 } from './_context'

const PropDrilling_02 = () => {
  return (
    <PeopleContextProvider_02>
      <Wrapper>
        <div className='container'>
          <h3>Context API Demo</h3>
          <List />
        </div>
      </Wrapper>
    </PeopleContextProvider_02>
  )
}

const List = () => {
  const { people } = usePeopleContext_02()
  return (
    <>
      {people.map((person) => {
        return <SinglePerson key={person.id} {...person} />
      })}
    </>
  )
}

const SinglePerson = ({ id, name }) => {
  const { removePerson } = usePeopleContext_02()
  return (
    <div className='item'>
      <h4>{name}</h4>
      <button className='btn' onClick={() => removePerson(id)}>Remove</button>
    </div>
  )
}
```

0f09928 apple550678

2025-12-14 15:20:51 +0800

Video: W13-P2: Context API demo for W13-

Video: W13-P3: Move W7 BlogJson2 page to the route /demo/blog_xx/localjson2_xx

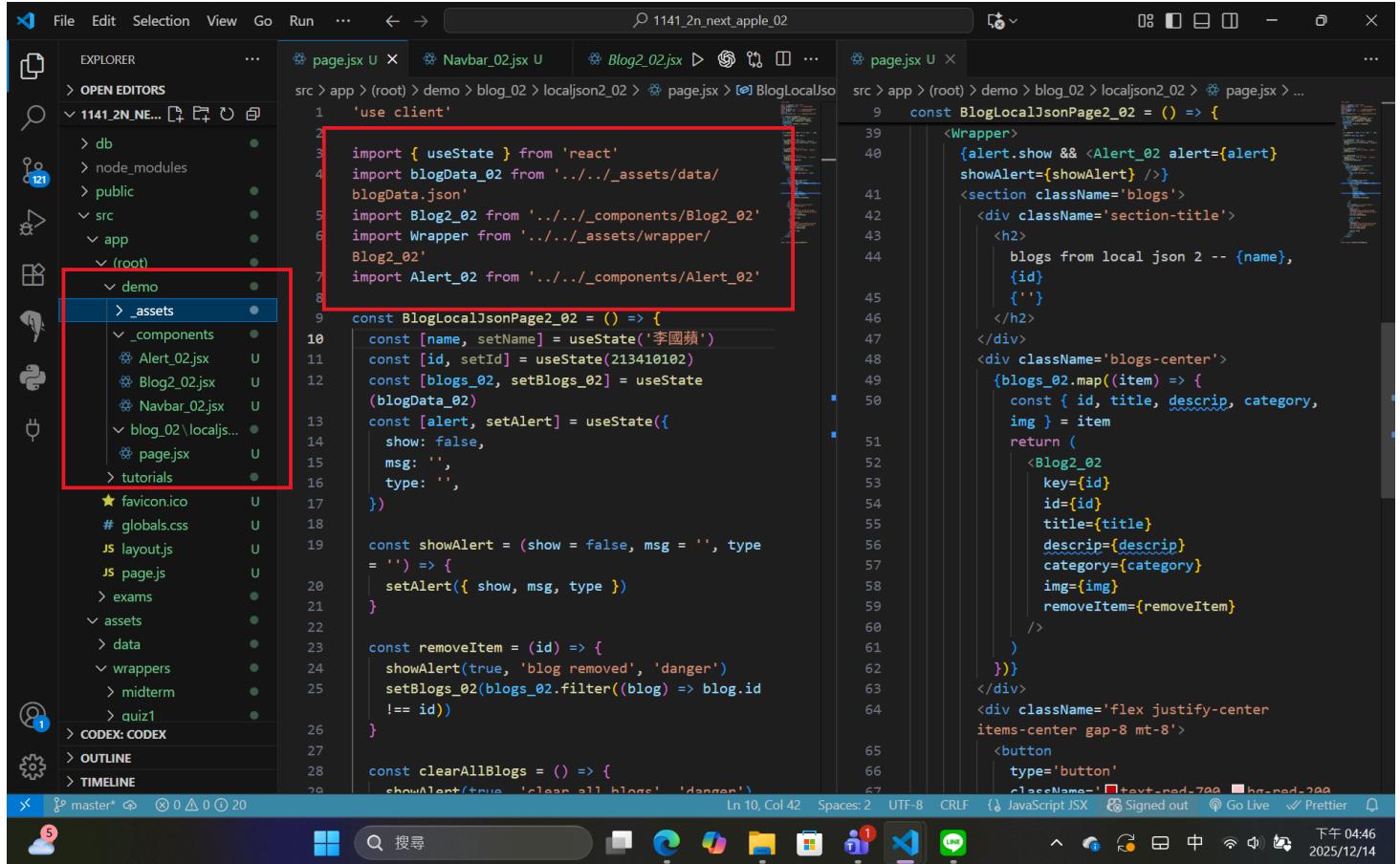
=> Chrome, show the result

The screenshot shows a web browser window with multiple tabs open. The active tab's URL is `localhost:3000/demo/blog_02/localjson2_02`. The page displays a grid of blog posts. A context menu is open over the third post in the first row, listing items like "W2-BookList_02_1", "W3-BlogStatic_02_1", etc., with "W7-BlogLocalJson2_02" highlighted and surrounded by a red box. The browser's status bar at the bottom shows the URL again and the date/time: "下午 04:42 2025/12/14".

LIFESTYLE	Seven Reasons Why Coffee Is Awesome	Seven Reasons Why Coffee Is Awesome
TRAVEL	Travel To Paris	Travel To Paris
Coffee Brings Friendship	Coffee Brings Friendship	Coffee Brings Friendship
Coffee Make You Feel Good	Coffee Make You Feel Good	Coffee Make You Feel Good
Coffee Make You Calm	Coffee Make You Calm	Coffee Make You Calm

Below the table, there are "Read More" and "Delete" buttons for each post.

=> relevant code



The screenshot shows a code editor interface with multiple tabs open. The current tab is 'BlogLocalJsonPage2_02.js'. The code within this tab is highlighted with a red box. The code itself is a component definition:

```
const BlogLocalJsonPage2_02 = () => {
  const [name, setName] = useState('李國蘋')
  const [id, setId] = useState(213410102)
  const [blogs_02, setBlogs_02] = useState(blogData_02)
  const [alert, setAlert] = useState({
    show: false,
    msg: '',
    type: ''
  })

  const showAlert = (show = false, msg = '', type = '') => {
    setAlert({ show, msg, type })
  }

  const removeItem = (id) => {
    showAlert(true, 'blog removed', 'danger')
    setBlogs_02(blogs_02.filter((blog) => blog.id !== id))
  }

  const clearAllBlogs = () => {
    showAlert(true, 'clear all blogs', 'danger')
  }
}
```

The code editor's sidebar shows a tree view of the project structure, with a red box highlighting the 'assets' folder under 'demo'. The status bar at the bottom shows the file path '1141_2n_next.apple.02', line 10, column 42, and other details.

e15f4de apple550678

2025-12-14 16:48:37 +0800

Video: W13-P3: Move W7 BlogJson2 page to