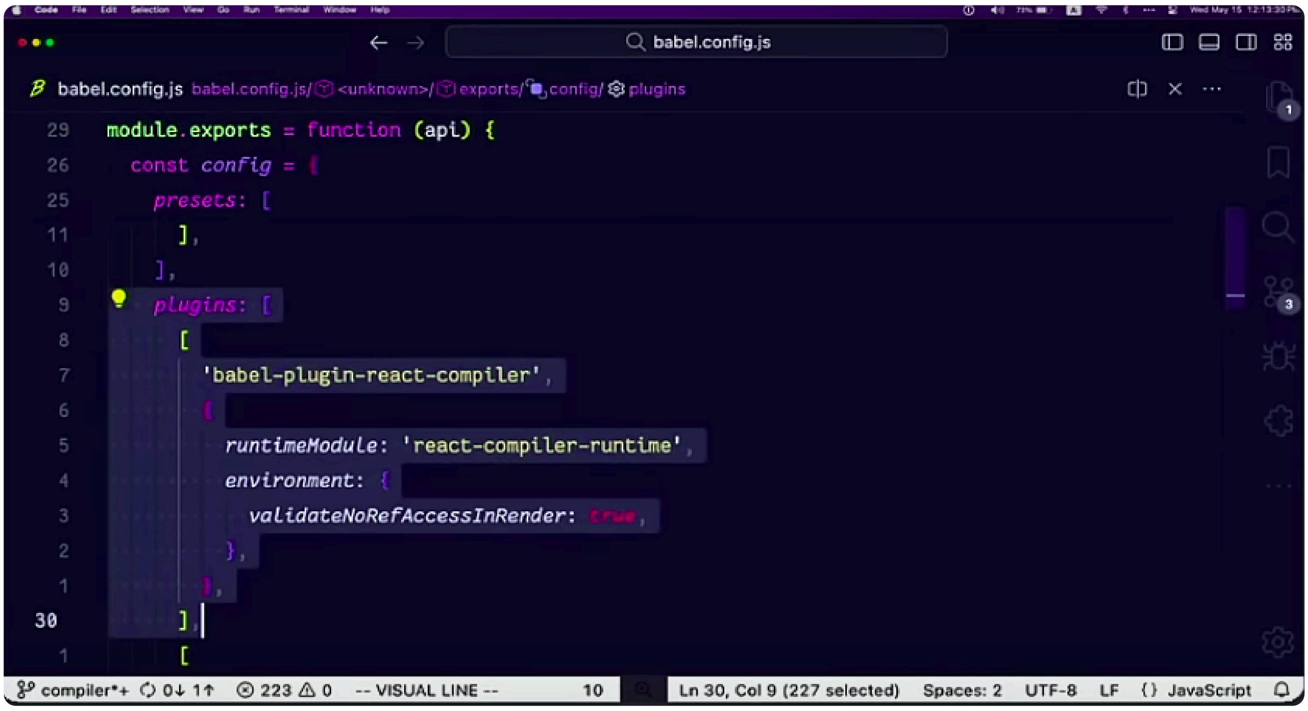
📑 5.Forget About Memo

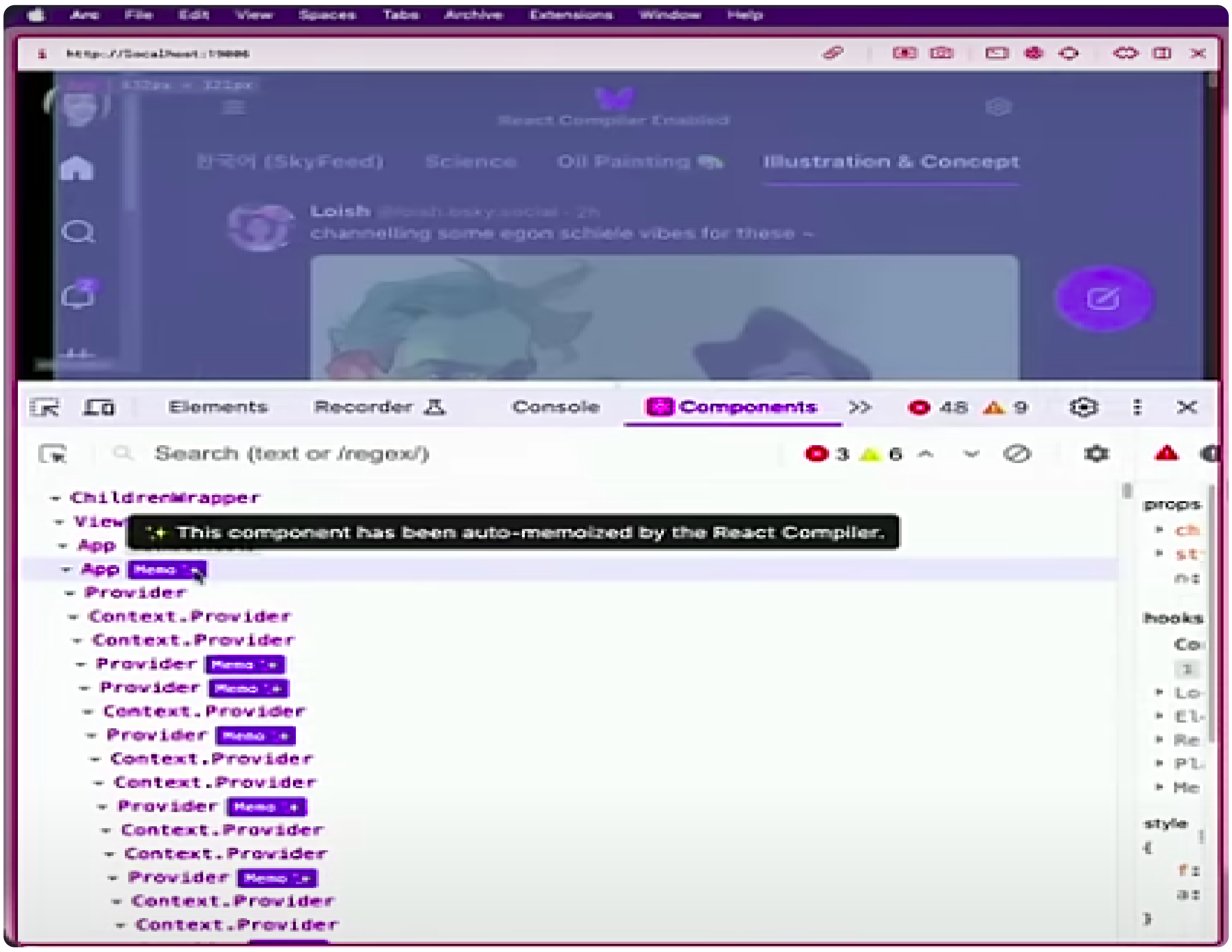
Demo



babel.config.js babel.config.js/<unknown>/exports/config/plugins

```
module.exports = function (api) {
  const config = {
    presets: [
      ],
    ],
    plugins: [
      [
        'babel-plugin-react-compiler',
        {
          runtimeModule: 'react-compiler-runtime',
          environment: {
            validateNoRefAccessInRender: true,
          },
        },
      ],
      [
```

compiler*+ 0↓ 1↑ ⊗ 223 ⚠ 0 -- VISUAL LINE -- 10 Ln 30, Col 9 (227 selected) Spaces: 2 UTF-8 LF {} JavaScript



http://localhost:19006

React Compiler Enabled

한국어 (SkyFeed) Science Oil Painting Illustration & Concept

Loish @loish.bsky.social · 2h
channelling some egon schiele vibes for these ~

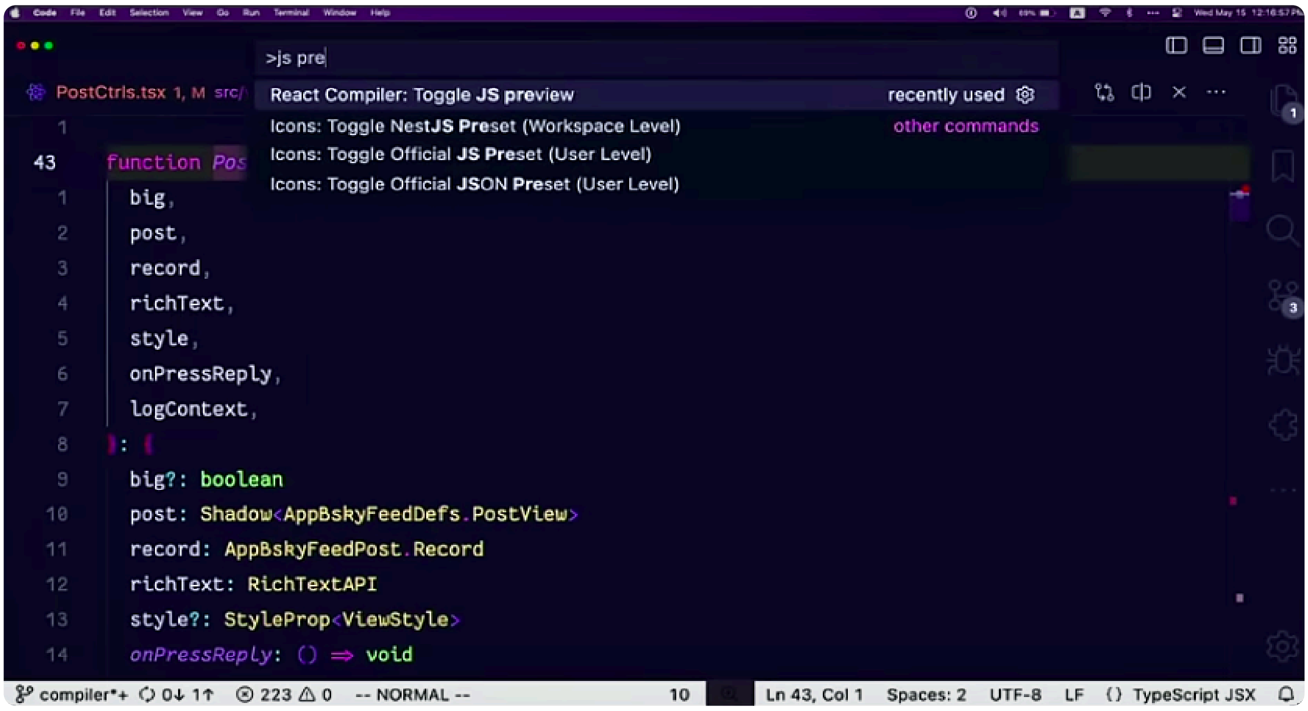
Elements Recorder Console Components 48 9

Search (text or /regex/) 3 6

```
▾ ChildrenWrapper
 ▾ View
  ▾ App
   ▾ App  Memo ✨
    ▾ Provider
     ▾ Context.Provider
      ▾ Context.Provider
       ▾ Provider  Memo ✨
        ▾ Provider  Memo ✨
         ▾ Context.Provider
          ▾ Provider  Memo ✨
           ▾ Context.Provider
            ▾ Context.Provider
             ▾ Provider  Memo ✨
              ▾ Context.Provider
               ▾ Context.Provider
                ▾ Provider  Memo ✨
                 ▾ Context.Provider
                  ▾ Context.Provider
```

✨ This component has been auto-memoized by the React Compiler.

props
hooks
style

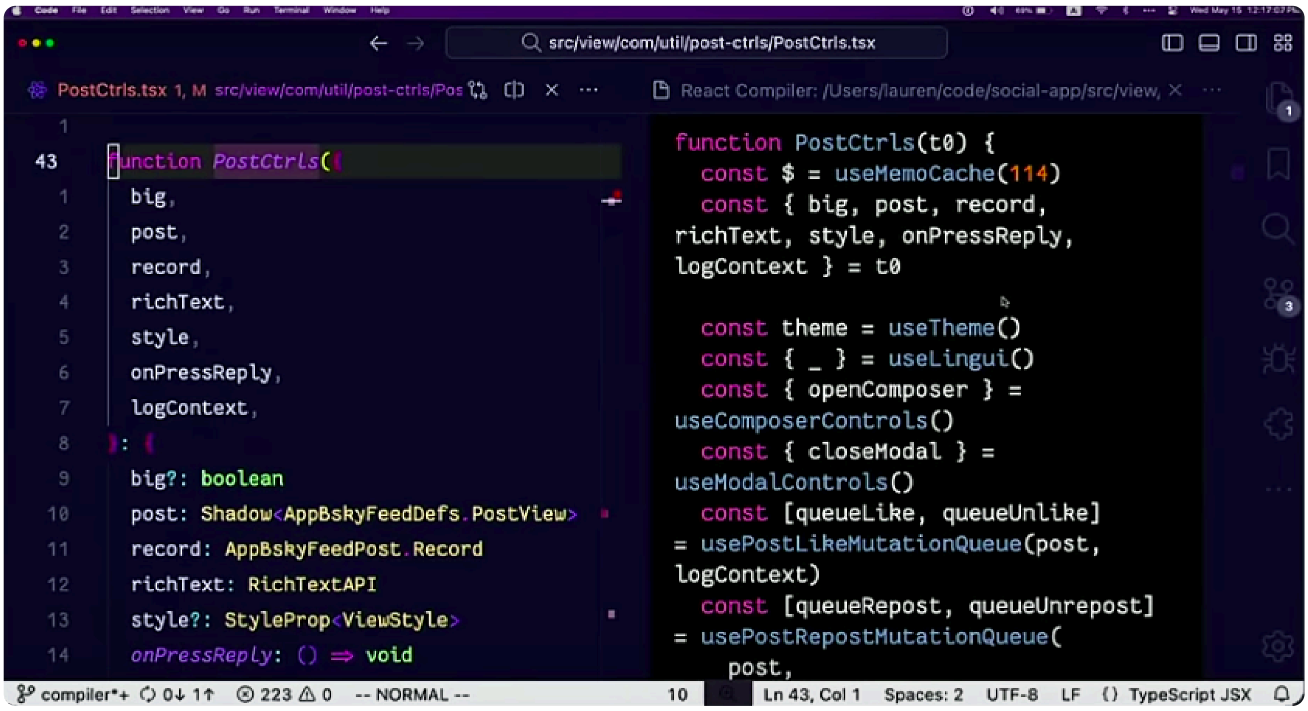


```
>js pre
React Compiler: Toggle JS preview                    recently used
Icons: Toggle NestJS Preset (Workspace Level)        other commands
Icons: Toggle Official JS Preset (User Level)
Icons: Toggle Official JSON Preset (User Level)
```

PostCtrls.tsx 1, M src/

```
function Pos
  big,
  post,
  record,
  richText,
  style,
  onPressReply,
  logContext,
}: {
  big?: boolean
  post: Shadow<AppBskyFeedDefs.PostView>
  record: AppBskyFeedPost.Record
  richText: RichTextAPI
  style?: StyleProp<ViewStyle>
  onPressReply: () => void
```

compiler*+ 0↓ 1↑ ⊗ 223 ⚠ 0 -- NORMAL -- 10 Ln 43, Col 1 Spaces: 2 UTF-8 LF {} TypeScript JSX



src/view/com/util/post-ctrls/PostCtrls.tsx

PostCtrls.tsx 1, M src/view/com/util/post-ctrls/Pos

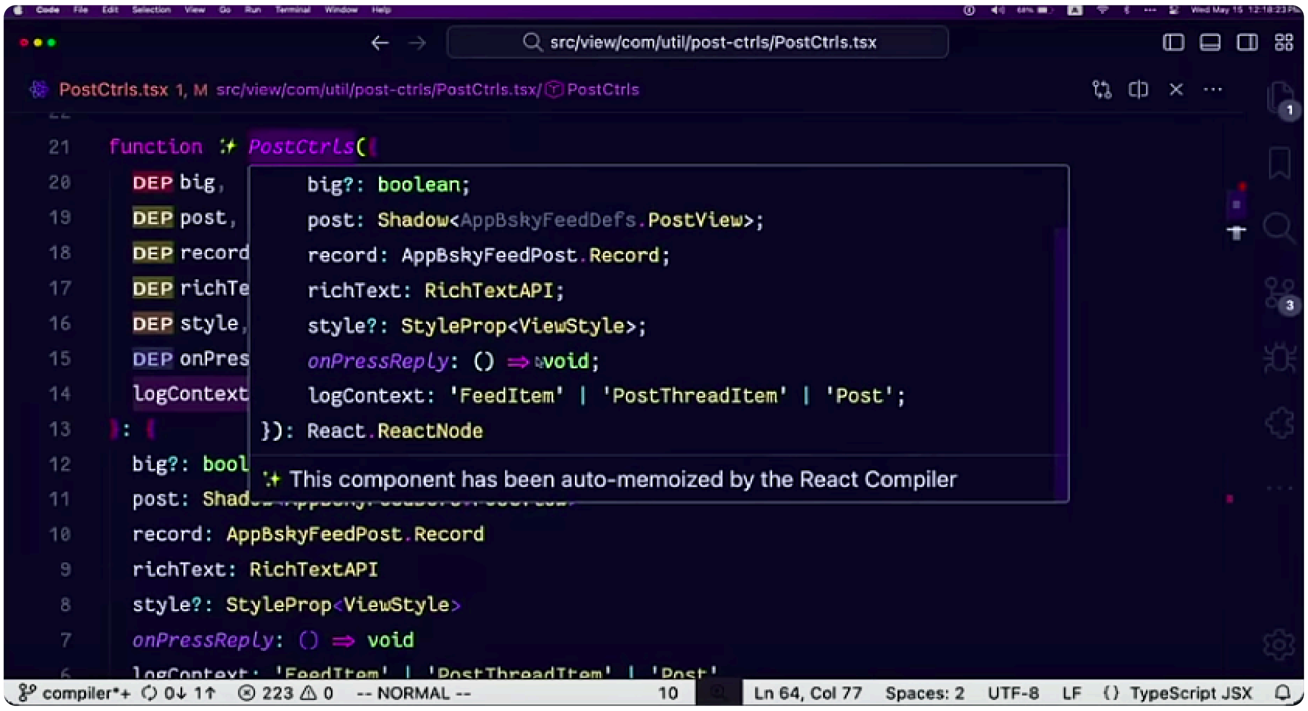
React Compiler: /Users/lauren/code/social-app/src/view,

```
function PostCtrls({
  big,
  post,
  record,
  richText,
  style,
  onPressReply,
  logContext,
}: {
  big?: boolean
  post: Shadow<AppBskyFeedDefs.PostView>
  record: AppBskyFeedPost.Record
  richText: RichTextAPI
  style?: StyleProp<ViewStyle>
  onPressReply: () => void
```

```
function PostCtrls(t0) {
  const $ = useMemoCache(114)
  const { big, post, record,
richText, style, onPressReply,
logContext } = t0

  const theme = useTheme()
  const { _ } = useLingui()
  const { openComposer } =
useComposerControls()
  const { closeModal } =
useModalControls()
  const [queueLike, queueUnlike]
= usePostLikeMutationQueue(post,
logContext)
  const [queueRepost, queueUnrepost]
= usePostRepostMutationQueue(
    post,
```

compiler*+ 0↓ 1↑ ⊗ 223 ⚠ 0 -- NORMAL -- 10 Ln 43, Col 1 Spaces: 2 UTF-8 LF {} TypeScript JSX

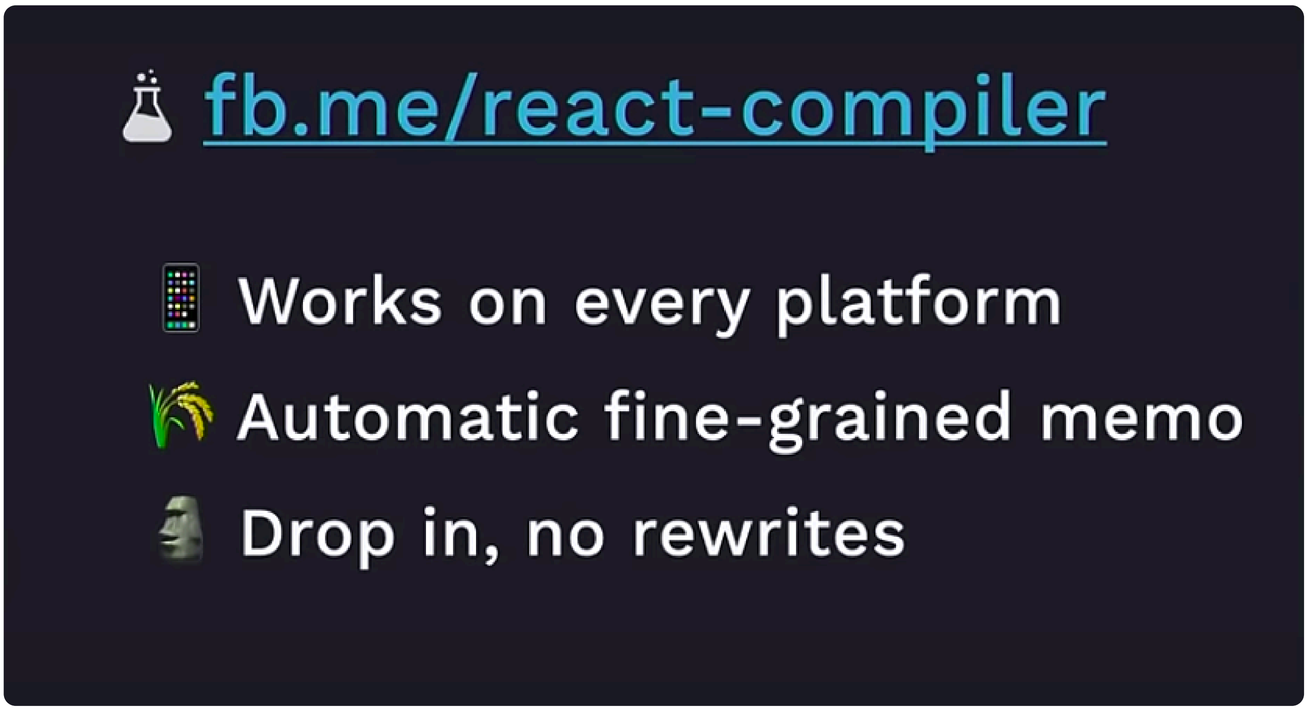

src/view/com/util/post-ctrls/PostCtrls.tsx
PostCtrls.tsx 1, M src/view/com/util/post-ctrls/PostCtrls.tsx/PostCtrls

```
function ✨ PostCtrls({
  DEP big,
  DEP post,
  DEP record
  DEP richTe
  DEP style,
  DEP onPres
  logContext
}: {
  big?: bool
  post: Shad
  record: AppBskyFeedPost.Record
  richText: RichTextAPI
  style?: StyleProp<ViewStyle>
  onPressReply: () => void
  logContext: 'FeedItem' | 'PostThreadItem' | 'Post'
```

```
    big?: boolean;
    post: Shadow<AppBskyFeedDefs.PostView>;
    record: AppBskyFeedPost.Record;
    richText: RichTextAPI;
    style?: StyleProp<ViewStyle>;
    onPressReply: () => void;
    logContext: 'FeedItem' | 'PostThreadItem' | 'Post';
}): React.ReactNode
```

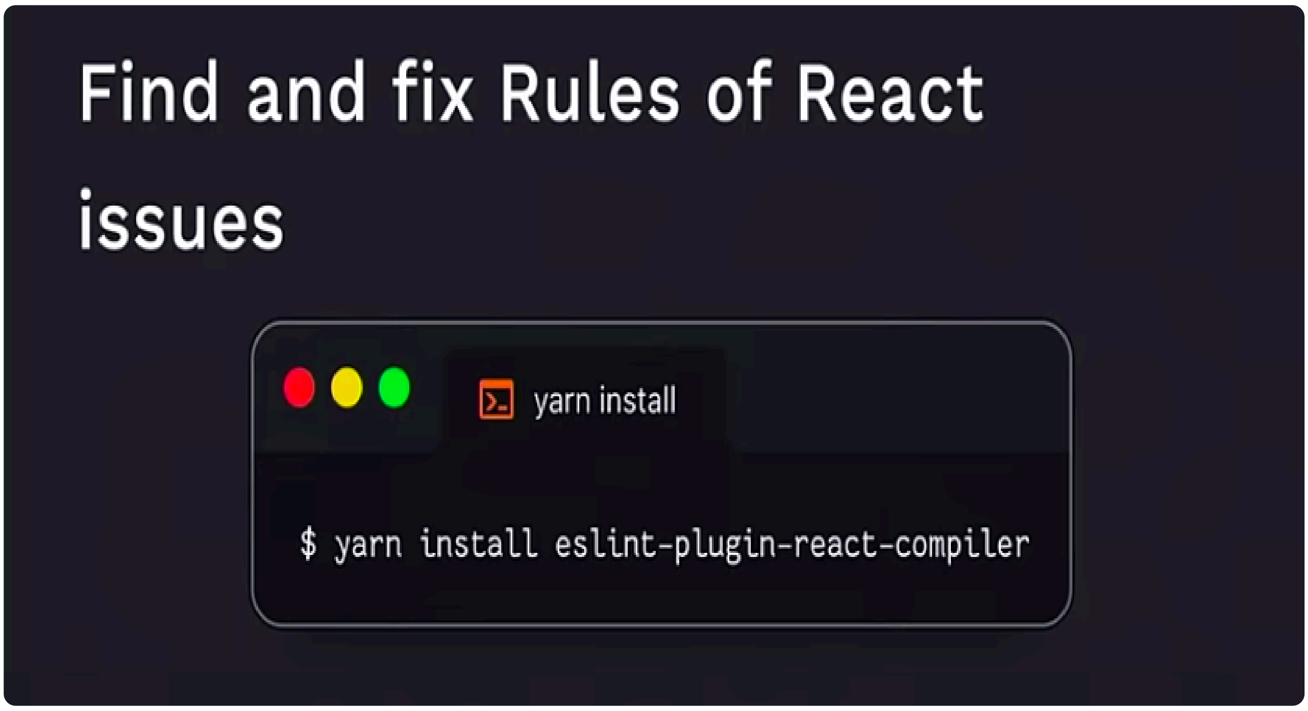
✨ This component has been auto-memoized by the React Compiler

compiler*+ 0↓ 1↑ ⊗ 223 ⚠ 0 -- NORMAL -- 10 Ln 64, Col 77 Spaces: 2 UTF-8 LF {} TypeScript JSX



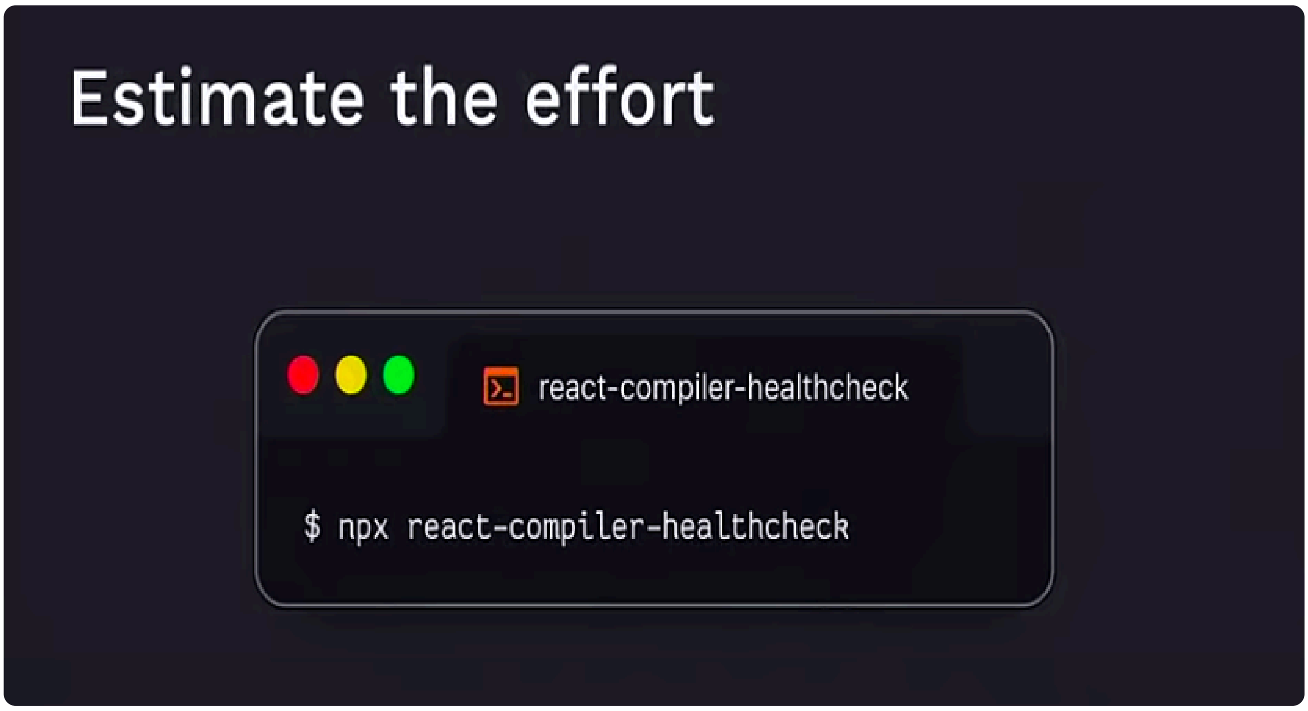
⚗️ fb.me/react-compiler

- 📱 Works on every platform
- 🌾 Automatic fine-grained memo
- 🗿 Drop in, no rewrites



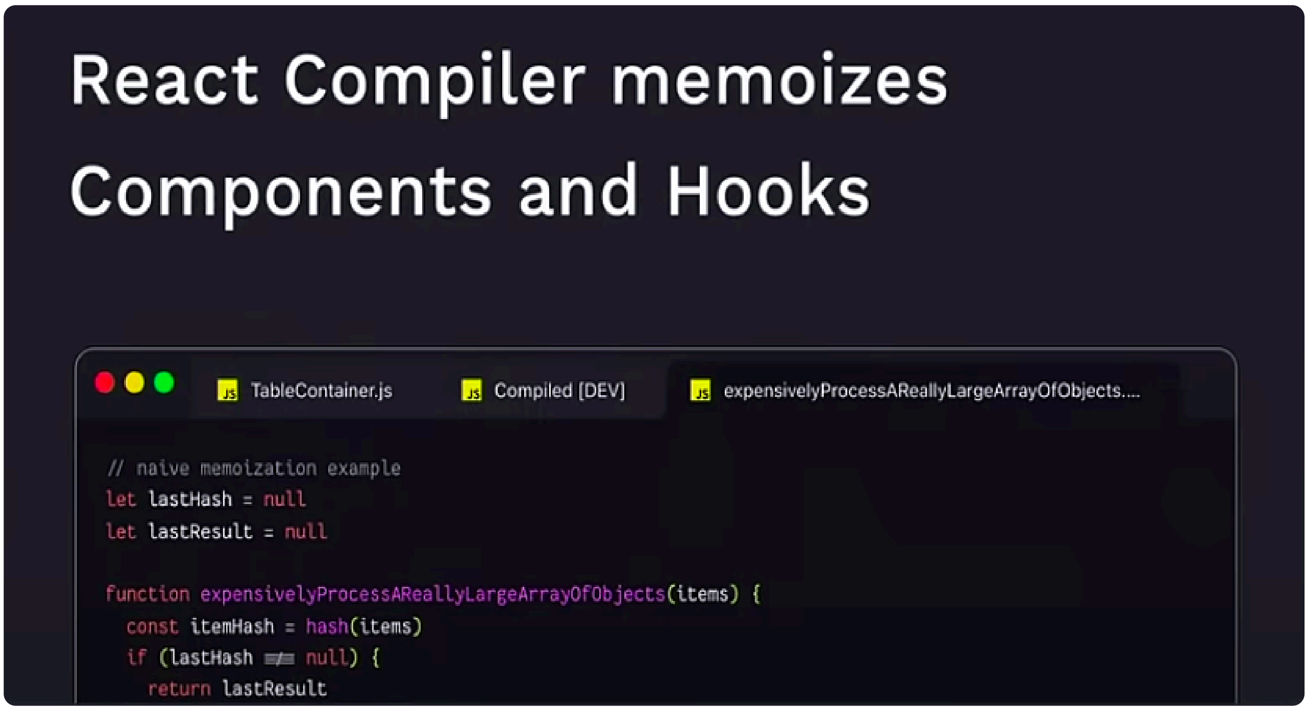
Find and fix Rules of React issues

```
$ yarn install eslint-plugin-react-compiler
```



Estimate the effort

```
$ npx react-compiler-healthcheck
```



React Compiler memoizes Components and Hooks

```
// naive memoization example
let lastHash = null
let lastResult = null

function expensivelyProcessAReallyLargeArrayOfObjects(items) {
  const itemHash = hash(items)
  if (lastHash !== null) {
    return lastResult
```