

@types/react

`@types/react` is the **npm package containing TypeScript definitions for the React library**, essential for building React apps with TypeScript to get type-checking for components, props, hooks (like `useState`, `useEffect`), and rendering types (like `ReactNode`). It provides the necessary structure for TypeScript to understand React's API, enabling features like autocompletion and catching errors before runtime, with types such as `React.FC` for functional components or `React.ReactNode` for renderable content.

What it does:

- **Type Safety:** Gives TypeScript knowledge of React's functions, components, and props.
- **Hooks Support:** Includes types for all built-in React hooks, allowing their correct usage.
- **Component Definition:** Helps define component props, including `children` (often as `React.ReactNode`).
- **Rendering Types:** Defines types like `React.Element`, `React.ReactNode` (anything renderable), `JSX.Element`, etc..

How to use it:

- Install it as a development dependency: `npm install --save-dev @types/react` or `yarn add @types/react`.
- It's typically installed alongside `@types/react-dom` for full type support in your React+TypeScript projects.

Key types you'll use:

- `React.FC<Props>`: For functional components.
- `React.ReactNode`: For props that accept anything renderable (children, content).
- `React.HTMLAttributes<T>`: For standard HTML attributes on components.

Important Note:

- Ensure your `@types/react` version matches your `react` version for consistency, as mismatches can cause issues (e.g., React 18 changes required corresponding type updates)

