

Using Hooks in React 18

Introduction



Peter Kellner

Developer, Consultant and Author

ReactAtScale.com @pkellner linkedin.com/in/peterkellner99

Using Hooks in React 18

Version Check



Version Check



This course was created by using React 18



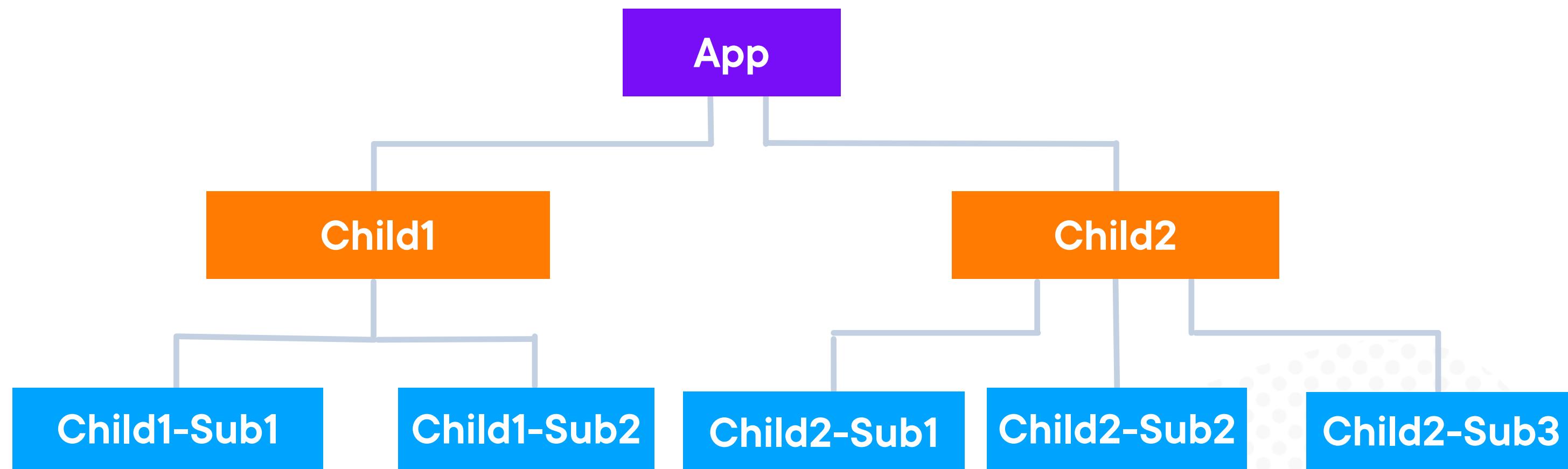
Version Check



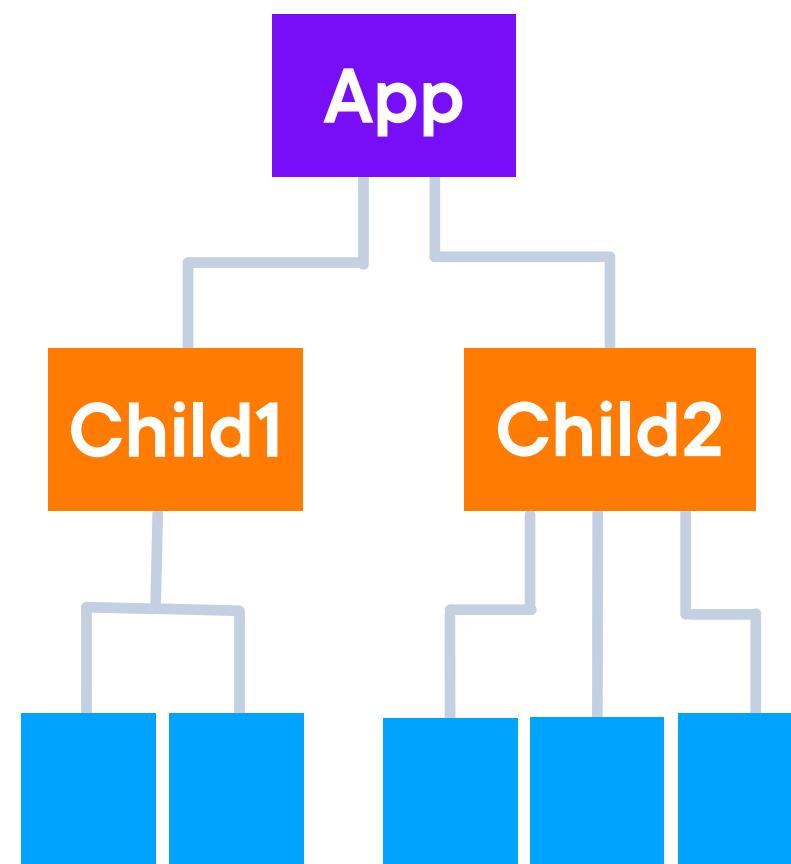
This course is 100% applicable to React 18



Component Hierarchy In a React App



Component Hierarchy In a React App



Component Hierarchy In a React App

```
export default function App() {  
  
  const helloMessage =  
    "Hello From Pluralsight";  
  
  return (  
    <>  
      <h1>{helloMessage}</h1>  
      <Child1 /><Child2 />  
    </>  
  );  
}
```



Component Hierarchy In a React App

```
import { useState } from "react";

export default function App() {

  const helloMessage =
    "Hello From Pluralsight";

  return (
    <>
      <h1>{helloMessage}</h1>
      <Child1 /><Child2 />
    </>
  );
}
```



Component Hierarchy In a React App

```
import { useState } from "react";

export default function App() {

  const [helloMessage, setHelloMessage] =
    useState("Hello From Pluralsight");

  return (
    <>
      <h1>{helloMessage}</h1>
      <Child1 /><Child2 />
    </>
  );
}
```



Component Hierarchy In a React App

```
import { useState, useEffect } from "react";

export default function App() {

  const [helloMessage, setHelloMessage] =
    useState("Hello From Pluralsight");

  return (
    <>
      <h1>{helloMessage}</h1>
      <Child1 /><Child2 />
    </>
  );
}
```



Component Hierarchy In a React App

```
import { useState, useEffect } from "react";

export default function App() {

  const [helloMessage, setHelloMessage] =
    useState("Hello From Pluralsight");

  return (
    <>
      <h1>{helloMessage}</h1>
      <Child1 /><Child2 />
    </>
  );
}
```



Component Hierarchy In a React App

```
import { useState, useEffect } from "react";

export default function App() {

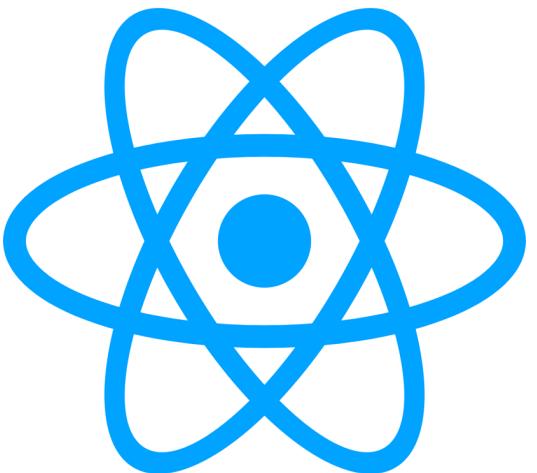
  const [helloMessage, setHelloMessage] =
    useState("Hello From Pluralsight");

  useEffect(() =>
    setHelloMessage
      (`${helloMessage} on ${new Date()}`), []
  );

  return (
    <>
      <h1>{helloMessage}</h1>
      <Child1 /><Child2 />
    </>
  );
}
```



React Hooks



**Reuse stateful logic
among components**



**Allow easy decoupling
of code logic from state
and lifecycle events**



**Better code
composition**



What to Know before Starting This Course



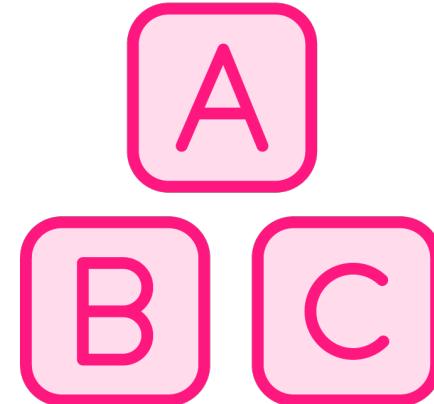
React Fundamentals

- Components
- JSX

Modern JavaScript

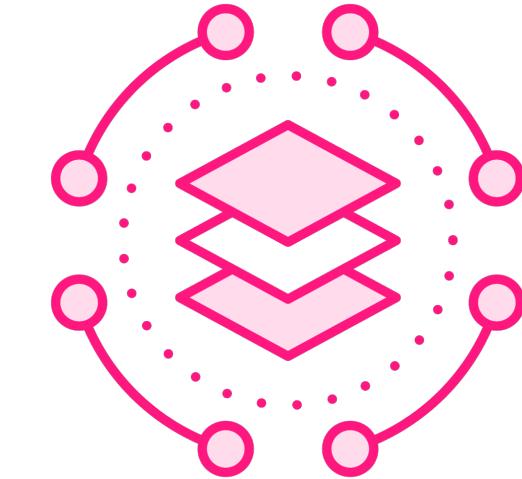


The 15 Hooks Built-in to React 18



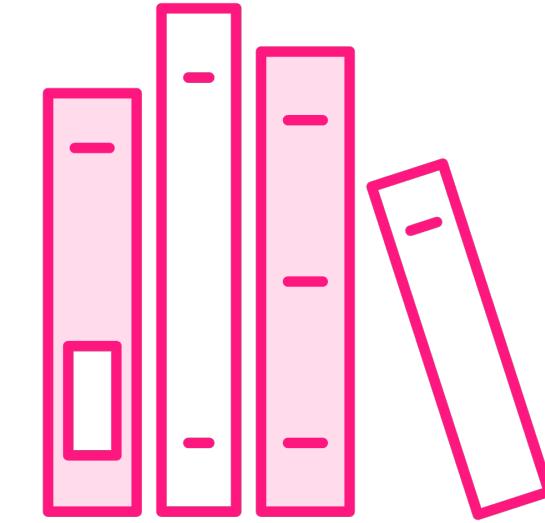
Basic Hooks

**useState, useEffect,
useContext**



Additional Hooks

**useReducer, useCallback,
useMemo, useRef,
useImperativeHandle,
useLayoutEffect,
useDebugValue,
useDeferredValue,
useTransition, useId**



Library Hooks

**useSyncExternalStore,
useInsertionEffect**



Up Next:

Understanding React Hooks

