HELLO!

I am Alfredo Alcántara

Frontend Developer

CONDITIONAL RENDERING

Create distinct components, then you can render only some of them, depending on the state of your application.

How can we do it?

▸ Conditionals operators (if-else)

▸ Inline if w/ embed expressions ({ } - &&)

▸ Inline if-else (condition ? true : false)

▸ State of the app (hooks)

Conditionals operators (if-else)

```
function Greeting(props) {
  const isLoggedIn = props.isLoggedIn;
  if (isLoggedIn) {
    return <UserGreeting />;
  }
  return <GuestGreeting />;
}
```

Inline if w/ embed expressions ({ } - &&)

```javascript
function Mailbox(props) {
  const unreadMessages = props.unreadMessages;
  return (
    <div>
      <h1>Hello!</h1>
      {unreadMessages.length > 0 &&
        <h2>
          You have {unreadMessages.length} unread messages.
        </h2>
      }
    </div>
  );
}

const messages = ['React', 'Re: React', 'Re:Re: React'];
ReactDOM.render(
  <Mailbox unreadMessages={messages} />,
  document.getElementById('root')
);
```

Inline if-else (condition ? true : false)

```
render() {
  const isLoggedIn = this.state.isLoggedIn;
  return (
    <div>
      The user is <b>{isLoggedIn ? 'currently' : 'not'}</b> logged in.
    </div>
  );
}
```

```
render() {
  const isLoggedIn = this.state.isLoggedIn;
  return (
    <div>
      {isLoggedIn
        ? <LogoutButton onClick={this.handleLogoutClick} />
        : <LoginButton onClick={this.handleLoginClick} />
      }
    </div>
  );
}
```
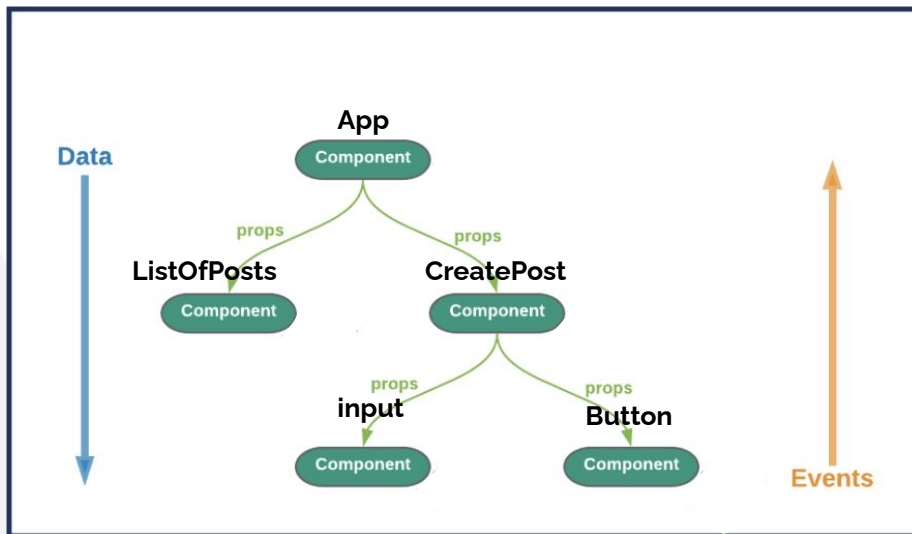
React Hooks

Control the state and other React features...

What is a hook and when to use it?

A Hook is a special function that lets you "hook into" React features.

Basic hooks

useState

```
import React, { useState } from 'react';

function Example() {
  const [count, setCount] = useState(0);

  return (
    <div>
      <p>You clicked {count} times</p>
      <button onClick={() => setCount(count + 1)}>
        Click me
      </button>
    </div>
  );
}
```

useEffect

```
useEffect(() => {
  document.title = `You clicked ${count} times`;
}, [count]); // Only re-run the effect if count changes
```

BOOTCAMP

THANKS!

Any questions?