

MERN PART-TIME
ONLINE

<React>

Intro to React

Functional Components

Functional vs. Class

Functional Components

Props

Prop It Up

Synthetic Events

Hooks

useState

Using State, Setting

Children

Components can also have children. Children of components are those element(s)/component(s) that are within the two "tags" of JSX elements.

Let's say we have the following `App.js`.

```
import React from 'react';
import './App.css';
import MyNewComponent from './components/MyNewComponent';

function App() {
  return (
    <div className="App">
      <MyNewComponent header={ "Header Prop" }>
        <h1>These are children</h1>
        <p>This is a child</p>
        <p>This is another child</p>
        <p>This is even another child</p>
      </MyNewComponent>
    </div>
  );
}
```

```
export default App;
```

We have a Component called `MyNewComponent`. There is one prop called `header` which we can access within our `MyNewComponent.js`. We also have some HTML elements between the opening and closing `MyNewComponent` JSX tags. All elements between these two tags are called "children". We can access them through props via the following:

MyNewComponent.js

```
import React from 'react';

const MyNewComponent = (props) => {
  return(
    <div>
      { props.children }
      <h1>
        { props.header }
      </h1>
    </div>
  );
}
```

```
export default MyNewComponent;
```

So, we can wrap content with JSX and then nest children so they can be accessed through props.

The following is the output from the code above:

These are children

This is a child

This is another child

This is even another child

Header Prop



