Default props

As we have said in the beginning of this Super Skill, props are a core idea in React's architecture. Let's reflect on something for a moment. If we have used a prop in a component but we forget to send its value from the parent component, that will inherently produce problems in the execution. For that reason, React provides a technique that will enormously help us. That technique is called the **Default props**.

This code defines a very simple ReactHeader component that renders an `<h1>` element containing a heading for the documentation of a particular React version.

```
// Simple React Component
function ReactHeader(props) {
 return <h1>React {props.version} Documentation</h1>;
}
```

In the ReactHeader component, we are going to set the prop version to 16. Everything else seems to be working properly in the ReactHeader component.

```
//We are going to render ReactHeader with version='16'
const  App = () => { return <ReactHeader version='16' />}
//output on the screen :
//React 16 Documentation
```

Default props

What happens when the version prop is not passed?

You probably might have guessed it right. Here is what happens when the ReactHeader component is rendered without the version prop:

```
// Simple React Component
function ReactHeader(props) {
```

```
 return <h1>React {props.version} Documentation</h1>;
}
```

Since the version prop is not passed, the reference to props.version in the component is undefined.

The output on the screen will be :

```
//output on the screen :
React undefined Documentation
```

Solution:

You could fix this by setting the `default props` in the component :

```
// Simple React Component
function ReactHeader(props) {
 return <h1>React {props.version} Documentation</h1>;
}

// Set default props
ReactHeader.defaultProps = {
 version: "16"
};
```

The component is going to use the **default value** for the version prop whenever it is not passed.

Default props with destructuring syntax

Alternatively, with the ES6 object destructuring syntax, you can destructure the props of a functional component with default values.

```
// Simple React Component
function ReactHeader(props) {
 // default value of version is 16
 const { version = "16" } = props;
 return <h1>React {version} Documentation</h1>;
}
```

```
//Or;

// default value of version is 16

function ReactHeader({ version = "16" }) {

 return <h1>React {version} Documentation</h1>;

}
```

The component is going to use the default value for the version prop whenever it is not passed.

‹ Previous

next ›