|  |  |
| --- | --- |
| **ch24.ReactJS Fragments** | **Date: 22-02-2022** |

**Topics**

React Props Validation,

# React Props Validation

# React Fragments

In React, whenever you want to render something on the screen, you need to use a render method inside the component. This render method can return **single** elements or **multiple** elements. The render method will only render a single root node inside it at a time. However, if you want to return multiple elements, the render method will require a '**div**' tag and put the entire content or elements inside it. This extra node to the DOM sometimes results in the wrong formatting of your HTML output and also not loved by the many developers.

### **Example**

1. // Rendering with div tag
2. **class** App **extends** React.Component {
3. render() {
4. **return** (
5. //Extraneous div element
6. <div>
7. <h2> Hello World! </h2>
8. <p> Welcome to the JavaTpoint. </p>
9. </div>
10. );
11. }
12. }

To solve this problem, React introduced **Fragments** from the **16.2** and above version. Fragments allow you to group a list of children without adding extra nodes to the DOM.

### **Syntax**

1. <React.Fragment>
2. <h2> child1 </h2>
3. <p> child2 </p>
4. .. ..... .... ...
5. </React.Fragment>

### **Example**

1. // Rendering with fragments tag
2. **class** App **extends** React.Component {
3. render() {
4. **return** (
5. <React.Fragment>
6. <h2> Hello World! </h2>
7. <p> Welcome to the JavaTpoint. </p>
8. </React.Fragment>
9. );
10. }
11. }

## **Why we use Fragments?**

The main reason to use Fragments tag is:

1. It makes the execution of code faster as compared to the div tag.
2. It takes less memory.

## **Fragments Short Syntax**

There is also another shorthand exists for declaring fragments for the above method. It looks like **empty** tag in which we can use of '<>' and '' instead of the '**React.Fragment**'.

### **Example**

1. //Rendering with short syntax
2. **class** Columns **extends** React.Component {
3. render() {
4. **return** (
5. <>
6. <h2> Hello World! </h2>
7. <p> Welcome to the JavaTpoint </p>
8. </>
9. );
10. }
11. }

## **Keyed Fragments**

The shorthand syntax does not accept key attributes. You need a key for mapping a collection to an array of fragments such as to create a description list. If you need to provide keys, you have to declare the fragments with the explicit <**React.Fragment**> syntax.

#### **Note: Key is the only attributes that can be passed with the Fragments.**

### **Example**

1. Function  = (props) {
2. **return** (
3. <Fragment>
4. {props.items.data.map(item => (
5. // Without the 'key', React will give a key warning
6. <React.Fragment key={item.id}>
7. <h2>{item.name}</h2>
8. <p>{item.url}</p>
9. <p>{item.description}</p>
10. </React.Fragment>
11. ))}
12. </Fragment>
13. )
14. }

Next Topic[React Router](https://www.javatpoint.com/react-router)