

ReactJS - Component

React component is the building block of a React application. Let us learn how to create a new React component and the features of React components in this chapter.

A React component represents a small chunk of user interface in a webpage. The primary job of a React component is to render its user interface and update it whenever its internal state is changed. In addition to rendering the UI, it manages the events belongs to its user interface. To summarize, React component provides below functionalities.

Initial rendering of the user interface.

Management and handling of events.

Updating the user interface whenever the internal state is changed.

React component accomplish these feature using three concepts –

Properties – Enables the component to receive input.

Events – Enable the component to manage DOM events and end-user interaction.

State – Enable the component to stay stateful. Stateful component updates its UI with respect to its state.

Let us learn all the concept one-by-one in the upcoming chapters.

Creating a React component

React library has two component types. The types are categorized based on the way it is being created.

Function component – Uses plain JavaScript function.

ES6 class component – Uses ES6 class.

The core difference between function and class component are –

Function components are very minimal in nature. Its only requirement is to return a *React element*.

```
function Hello() {  
  return '<div>Hello</div>'  
}
```

The same functionality can be done using ES6 class component with little extra coding.

```
class ExpenseEntryItem extends React.Component {
  render() {
    return (
      <div>Hello</div>
    );
  }
}
```

Class components supports state management out of the box whereas function components does not support state management. But, React provides a hook, *useState()* for the function components to maintain its state.

Class component have a life cycle and access to each life cycle events through dedicated callback apis. Function component does not have life cycle. Again, React provides a hook, *useEffect()* for the function component to access different stages of the component.

Creating a class component

Let us create a new React component (in our expense-manager app), *ExpenseEntryItem* to showcase an expense entry item. Expense entry item consists of name, amount, date and category. The object representation of the expense entry item is –

```
{
  'name': 'Mango juice',
  'amount': 30.00,
  'spend_date': '2020-10-10'
  'category': 'Food',
}
```

Open *expense-manager* application in your favorite editor.

Next, create a file, *ExpenseEntryItem.css* under *src/components* folder to style our component.

Next, create a file, *ExpenseEntryItem.js* under *src/components* folder by extending *React.Component*.

```
import React from 'react';
import './ExpenseEntryItem.css';
class ExpenseEntryItem extends React.Component {
}
```

Next, create a method *render* inside the *ExpenseEntryItem* class.

```
class ExpenseEntryItem extends React.Component {
  render() {
  }
}
```

Next, create the user interface using JSX and return it from *render* method.

```
class ExpenseEntryItem extends React.Component {
  render() {
    return (
      <div>
        <div><b>Item: </b> <em>Mango Juice</em></div>
        <div><b>Amount: </b> <em>30.00</em></div>
        <div><b>Spend Date: </b> <em>2020-10-10</em></div>
        <div><b>Category: </b> <em>Food</em></div>
      </div>
    );
  }
}
```

Next, specify the component as default export class.

```
import React from 'react';
import './ExpenseEntryItem.css';

class ExpenseEntryItem extends React.Component {
  render() {
    return (
      <div>
        <div><b>Item: </b> <em>Mango Juice</em></div>
        <div><b>Amount: </b> <em>30.00</em></div>
        <div><b>Spend Date: </b> <em>2020-10-10</em></div>
        <div><b>Category: </b> <em>Food</em></div>
      </div>
    );
  }
}
export default ExpenseEntryItem;
```

Now, we successfully created our first React component. Let us use our newly created component in *index.js*.

```
import React from 'react';
import ReactDOM from 'react-dom';
import ExpenseEntryItem from './components/ExpenseEntryItem'
```

```
ReactDOM.render(  
  <React.StrictMode>  
    <ExpenseEntryItem />  
  </React.StrictMode>,  
  document.getElementById('root')  
)  
);
```

Example

The same functionality can be done in a webpage using CDN as shown below –

```
<!DOCTYPE html>  
<html>  
  <head>  
    <meta charset="UTF-8" />  
    <title>React application :: ExpenseEntryItem component</title>  
  </head>  
  <body>  
    <div id="react-app"></div>  
  
    <script src="https://unpkg.com/react@17/umd/react.development.js" crossorigin></script>  
    <script src="https://unpkg.com/react-dom@17/umd/react-dom.development.js" crossorigin></script>  
    <script src="https://unpkg.com/@babel/standalone/babel.min.js"></script>  
    <script type="text/babel">  
      class ExpenseEntryItem extends React.Component {  
        render() {  
          return (  
            <div>  
              <div><b>Item: </b> <em>Mango Juice</em></div>  
              <div><b>Amount: </b> <em>30.00</em></div>  
              <div><b>Spend Date: </b> <em>2020-10-10</em></div>  
              <div><b>Category: </b> <em>Food</em></div>  
            </div>  
          );  
        }  
      }  
      ReactDOM.render(  
        <ExpenseEntryItem />,  
        document.getElementById('react-app') );  
    </script>  
  </body>  
</html>
```

Next, serve the application using npm command.

```
npm start
```

Output

Next, open the browser and enter `http://localhost:3000` in the address bar and press enter.

```
Item: Mango Juice  
Amount: 30.00  
Spend Date: 2020-10-10  
Category: Food
```

Creating a function component

React component can also be created using plain JavaScript function but with limited features. Function based React component does not support state management and other advanced features. It can be used to quickly create a simple component.

The above `ExpenseEntryItem` can be rewritten in function as specified below –

```
function ExpenseEntryItem() {  
  return (  
    <div>  
      <div><b>Item: </b> <em>Mango Juice</em></div>  
      <div><b>Amount: </b> <em>30.00</em></div>  
      <div><b>Spend Date: </b> <em>2020-10-10</em></div>  
      <div><b>Category: </b> <em>Food</em></div>  
    </div>  
  );  
}
```

Here, we just included the render functionality and it is enough to create a simple React component.

Revision #1

Created 14 December 2022 10:39:34 by Admin

Updated 14 December 2022 10:40:02 by Admin