14. REACT Animations

In this chapter, we will learn how to animate elements using React.

14.1. Install React CSS Transitions Group (step 1)

This is React add-on used for creating basic CSS transitions and animations. We will install it from the command prompt window:

Example:

```
C:\Users\MyUser\Desktop\reactApp>npm install react-addons-css-
transition-group
```

14.2. Add a CSS file (step 2)

Let's create a new file style.css.

Example:

```
C:\Users\MyUser\Desktop\reactApp>type nul > css/style.css
```

To be able to use it in the app, we need to link it to the head element in index.html.

Example:

```
<!DOCTYPE html>
<html lang = "en">
   <head>
      <link rel = "stylesheet" type = "text/css" href = "./style.css">
      <meta charset = "UTF-8">
```

```
        <title>React App</title>
    </head>
    <body>
        <div id = "app"></div>
        <script src = 'index_bundle.js'></script>
    </body>
</html>
```

14.3. Appear Animation (step 3)

We will create a basic React component. The **ReactCSSTransitionGroup** element will be used as a wrapper of the component we want to animate. It will use *transitionAppear and transitionAppearTimeout, while transitionEnter and transitionLeave* are false.

Example: App.jsx

```
import React from 'react';
var ReactCSSTransitionGroup = require('react-addons-css-
transition-group');

class App extends React.Component {
    render() {
        return (
            <div>
                <ReactCSSTransitionGroup transitionName =
"example"
                    transitionAppear = {true}
transitionAppearTimeout = {500}
                    transitionEnter = {false} transitionLeave =
{false}>
```

```
            <h1>My Element...</h1>
          </ReactCSSTransitionGroup>
      </div>
    );
  }
}
export default App;
```

Example: main.js

```
import React from 'react'
import ReactDOM from 'react-dom';
import App from './App.jsx';


ReactDOM.render(<App />, document.getElementById('app'));
```
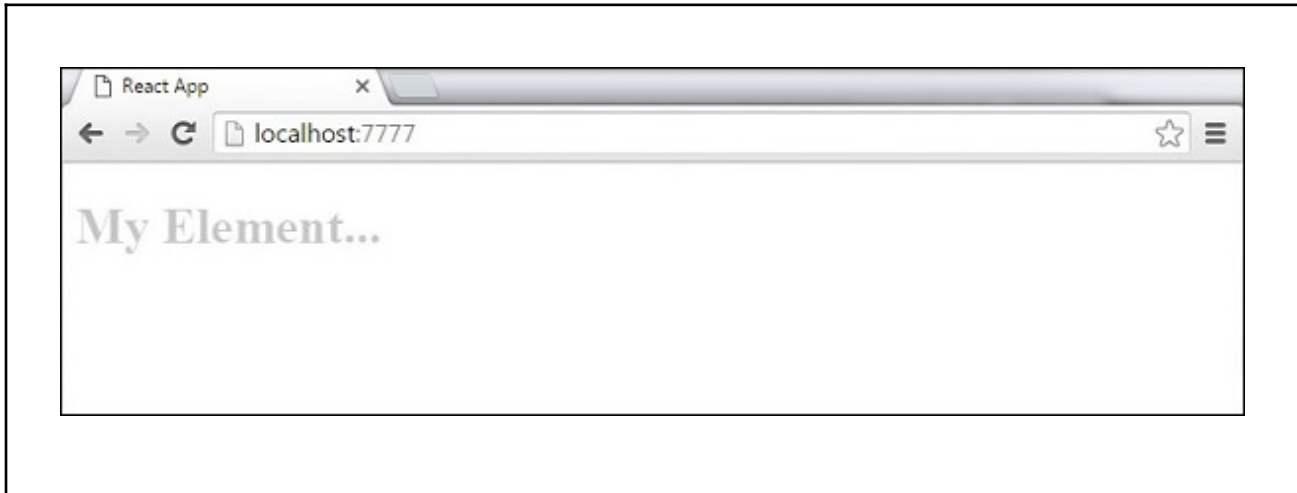
The CSS animation is very simple.

Example: css/style.css

```
.example-appear {
   opacity: 0.04;
}
.example-appear.example-appear-active {
   opacity: 2;
   transition: opacity 50s ease-in;
}
```

Once we start the app, the element will fade in.

Output:

14.4. Enter and Leave Animations (step 4)

Enter and leave animations can be used when we want to add or remove elements from the list.

Example: App.jsx

```jsx
import React from 'react';
var ReactCSSTransitionGroup = require('react-addons-css-
transition-group');

class App extends React.Component {
   constructor(props) {
      super(props);

      this.state = {
   items: ['Item 1...', 'Item 2...', 'Item 3...', 'Item 4...']
      }
      this.handleAdd = this.handleAdd.bind(this);
    };
```

```
    handleAdd() {

        var newItems = this.state.items.concat([prompt('Create
New Item')]);

        this.setState({items: newItems});

    }

    handleRemove(i) {

        var newItems = this.state.items.slice();

        newItems.splice(i, 1);

        this.setState({items: newItems});

    }

    render() {

        var items = this.state.items.map(function(item, i) {

            return (

                <div key = {item} onClick =
{this.handleRemove.bind(this, i)}>

                    {item}

                </div>

            );

        }.bind(this));


        return (

            <div>

             <button onClick = {this.handleAdd}>Add Item</button>


             <ReactCSSTransitionGroup transitionName = "example"

                    transitionEnterTimeout = {500}
transitionLeaveTimeout = {500}>

                    {items}

                </ReactCSSTransitionGroup>

            </div>

        );

    }
```

```
    }
    export default App;
```

Example: main.js

```
    import React from 'react'
    import ReactDOM from 'react-dom';
    import App from './App.jsx';


    ReactDOM.render(<App />, document.getElementById('app'));
```

Example: css/style.css

```
    .example-enter {
        opacity: 0.04;
    }
    .example-enter.example-enter-active {
        opacity: 5;
        transition: opacity 50s ease-in;
    }
    .example-leave {
        opacity: 1;
    }
    .example-leave.example-leave-active {
        opacity: 0.04;
        transition: opacity 50s ease-in;
    }
```
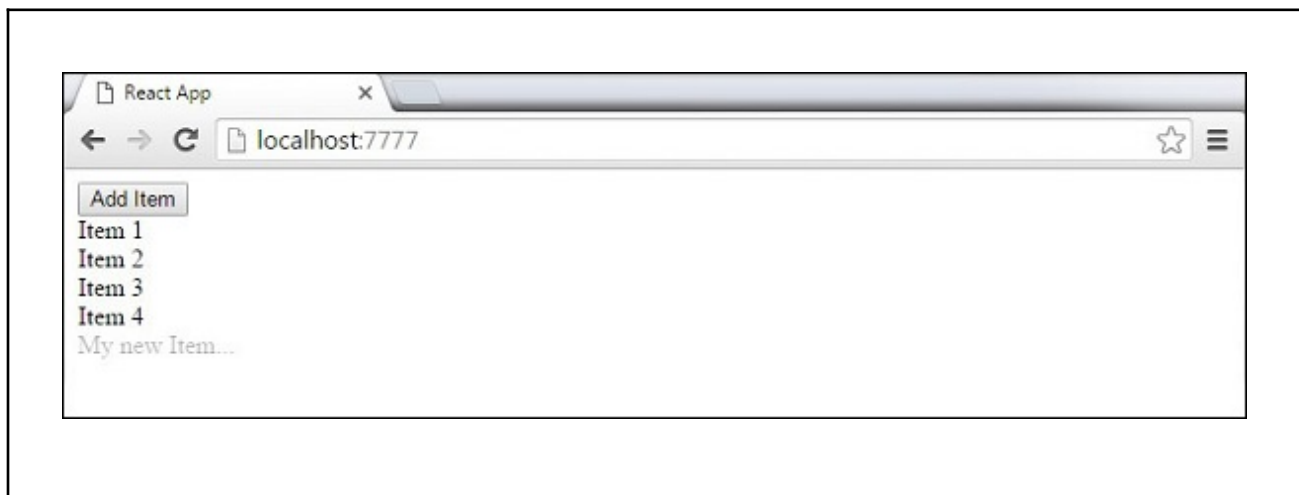
When we start the app and click the Add Item button, the prompt will appear.

Output:
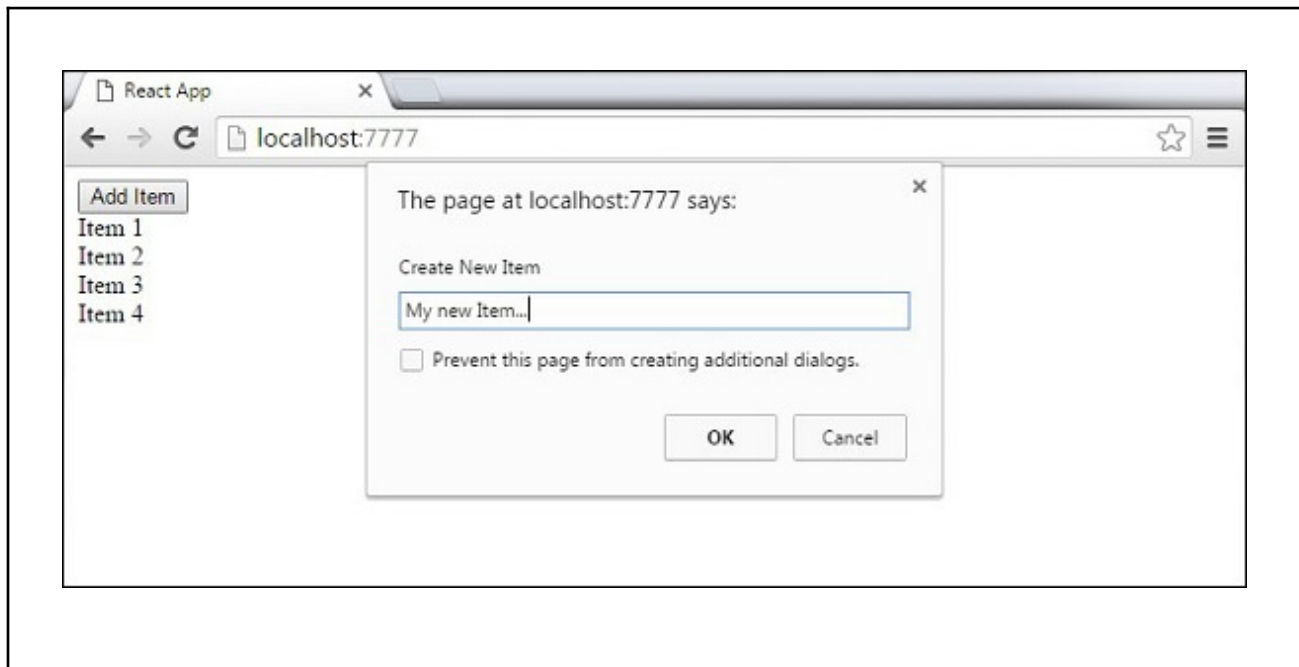
Once we enter the name and press OK, the new element will fade in.

Output:

Now we can delete some of the items (Item 3...) by clicking it. This item will fade out from the list.

Output: