

Elements.

In React, the smallest building blocks are the elements. Example of an element is -

```
const login = <h1> Welcome, you have successfully logged in! </h1>;
```

React DOM updates the actual DOM with the converted react elements. React components are made up of elements.

Rendering element on DOM

We will have a parent div element in the main .html file . This div can be called as root.

```
<div id="app"> </div>
```

ReactDOM manages everything which is inside the app div. We can add multiple such an isolated div in applications if required. To render the element it will be passed to the ReactDOM render method -

```
const login = <h1> Welcome, you have successfully logged in! </h1>;  
ReactDOM.render(login, document.getElementById( 'app' ));
```

This will display a message - Welcome, you have successfully logged in!

React elements are immutable that means once created it cannot be changed. Change will create a new element and update the UI.

Note that:

- An element describes what you want to see on the screen.
- React elements are plain objects, and are cheap to create. React DOM takes care of updating the DOM to match the React elements. bthis is very different from browser DOM elements.

React elements are immutable. Once you create an element, you can't change its children or attributes. An element is like a single frame in a movie: it represents the UI at a certain point in time.

With our knowledge so far, the only way to update the UI is to create a new element, and pass it to `ReactDOM.render()`.

- **In practice, most React apps only call `ReactDOM.render()`**

once.

React DOM compares the element and its children to the previous one, and only applies the DOM updates necessary to bring the DOM to the desired state.

Best Wishes Lux Tech Academy