|  |  |
| --- | --- |
| **ch41.ReactJS Loop Array in React JS** | **Date: 22-02-2022** |

**Topics**

React Props Validation,

# React Props Validation

# Loop Array in React JS | React Foreach Loop Example

In this section, we are going to use an array to explain the reach native loop in the render. For this, we have described a react loop array example in the render. When we want to build any web application, it is very important to have the knowledge of handling an array of data. In this example, we will see the use of loop in [react js](https://www.javatpoint.com/reactjs-tutorial)

. In our given example, we are going to use for loop in react js. To perform this, we have to follow some steps.

In this section, we will use react app. When we require a map, foreach loop, and for loop in react, we can look at the following example to learn the use of loop array n react js. In array, we always want for loop and foreach loop. When we want to loop our array in reach, it will require a map to do this. So we are going to explain the example of a map in react native. A new array is created by the map() method. In the calling array, it provides the result of calling a function on each and every element. The looping process can be simplified by this. When we use the map, we don't require the use of forEach function and for loop. Map, forEach loop, for loop has many differences. Instead of overwriting the existing data, the map function uses the data and creates a new array. Due to all the features and the simplicity of the map function, React docs strongly encourage us to use the map function.

We will provide two examples in react app to explain this easy concept. In the first example, we are going to describe the react loop, which has a single-dimensional array. In the second example, we will describe a loop that has a multidimensional array. Both examples are as follows:

**Example 1:**

20.4M

702

How to Install Unity on Ubuntu 18.04 [Complete Procedure]

**rc/App.js**

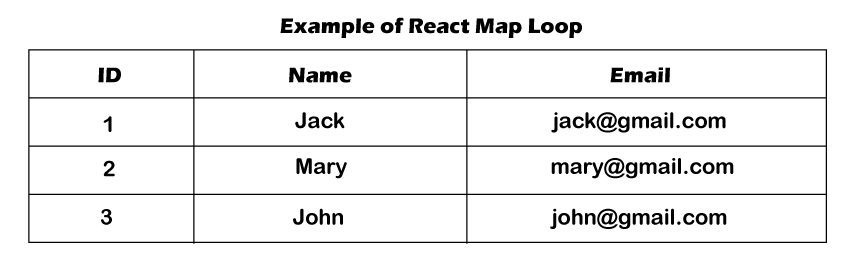
1. **import** React from 'react';
3. function App() {
5. **const** myArray = ['Jack', 'Mary', 'John', 'Krish', 'Navin'];
7. **return** (
8. <div className="container">
9. <h1> Example of React Map Loop </h1>
11. {myArray.map(name => (
12. <li>
13. {name}
14. </li>
15. ))}
17. </div>
18. );
19. }
21. export **default** App;

**Example 2:**

**src/App.js**

1. **import** React from 'react';
3. function App() {
5. **const** students = [
6. {
7. 'id': 1,
8. 'name': Jack,
9. 'email': 'jack@gmail.com'
10. },
11. {
12. 'id': 2,
13. 'name': 'Mary',
14. 'email': ?mary@gmail.com'
15. },
16. {
17. 'id': 3,
18. 'name': 'John',
19. 'email': 'john@gmail.com'
20. },
21. ];
23. **return** (
24. <div className="container">
25. <h1> Example of React Map Loop </h1>
27. <table className="table table-bordered">
28. <tr>
29. <th>ID</th>
30. <th>Name</th>
31. <th>Email</th>
32. </tr>
34. {students.map((student, index) => (
35. <tr data-index={index}>
36. <td>{student.id}</td>
37. <td>{student.name}</td>
38. <td>{student.email}</td>
39. </tr>
40. ))}
42. </table>
44. </div>
45. );
46. }
48. export **default** App;

After running this example, we will get the following preview:



Next Topic[React Axios Delete Request Example](https://www.javatpoint.com/react-axios-delete-request-example)