**KEYS IN REACT**

**What is a key in React?**

A “key” is a special string attribute you need to include when creating lists of elements in React. Keys are used in React to identify which items in the list are changed, updated, or deleted.

Keys are used to give an identity to the elements in the lists. It is recommended to use a string as a key that uniquely identifies the items in the list.

**Assigning keys to the list**

You can assign the array indexes as keys to the list items. The below example assigns array indexes as keys to the elements.

**Syntax:**

const numbers = [1, 2, 3, 4, 5];  
const updatedNums = numbers.map((number, index) =>  
 <li key={index}>  
 {number}  
 </li>  
);

**Issue with index as keys**

Assigning indexes as keys is **highly discouraged** because if the elements of the arrays get reordered in the future, then it will get confusing for the developer as the keys for the elements will also change.

## ****Using Keys with Components:****

Consider a situation where you have created a separate component for list items, and you are extracting list items from that component. In that case, you will have to assign keys to the component you are returning from the iterator and not to the list items. To avoid mistakes keep in mind that anything you are returning from inside of the map() function is needed to be assigned a key.

## ****Uniqueness of Keys:****

## Keys assigned to the array elements must be unique. However, this does not mean that the keys should be globally unique. All the elements in a particular array should have unique keys. That is, two different arrays can have the same set of keys.