

## Lec 3

### Pollyfill

- to make older browsers understand our new code, the new code is converted into a older code which browser can understand called pollyfill.
- babel do this conversion automatically.

Eg:- ES6 is the newer version of javascript.

If I'm working on 1999 browser,  
my browser will not understand what is this  
const, new Promise etc.

So, there is a replacement for code for these functionalities which is compaetible with older version of browsers.

- So, this is what happen when we write  
"browserslist" → our code is converted to older one.

### Babel

- is a javascript package / library used to convert code written in newer versions of JS (ECMAScript 2015, 2016, 2017 etc) into code that can be run in older JS engines.

Render - means updating something in the DOM.

### Reconciliation

- helps to make React applications fast and efficient by minimizing the amount of work that needs to be done to update the changes.
- So, you don't have to worry about what changes on every update.

- render() function as creating a tree of React elements.

- On the next state or props update, render() fn will return a different tree of React elements.

Whenever react is updating the DOM, for eg :-

<ul>

<li> Duke </li>

<li> Villanova </li>

</ul>

Now, I introduced one child over the top, then react will have to do lot of efforts, react will have to re-render everything. That means, [react will have to change the whole DOM tree.]

<ul>

<li> Connecticut </li>

<li> Duke </li>

<li> Villanova </li>

As react has to re-render everything, it will not give you good performance.



In large-scale application, it is far too expensive.

### SOLUTION - Introduction of Keys

- React supports 'key' attribute.
- When children have keys, React uses the key to match # children in the original tree with children in subsequent tree. Thus, making tree-conversion efficient.

<ul>

<li key="2014">Connecticut </li>

<li key="2015">Duke </li>

<li key="2016">Villanova </li>

</ul>

Thus, react has to do very less work.

So, always use keys whenever you have multiple children.

→ JSX is created to empower React.

### Advantages of JSX

- Developer experience
- Syntactical sugar
- Readability
- Less code
- maintainability
- No Repeation.