

Hooks, Props and State



Roland Guijt

Freelance consultant and trainer

@rolandguijt roland.guijt@gmail.com



Props

```
const Component = () =>
```

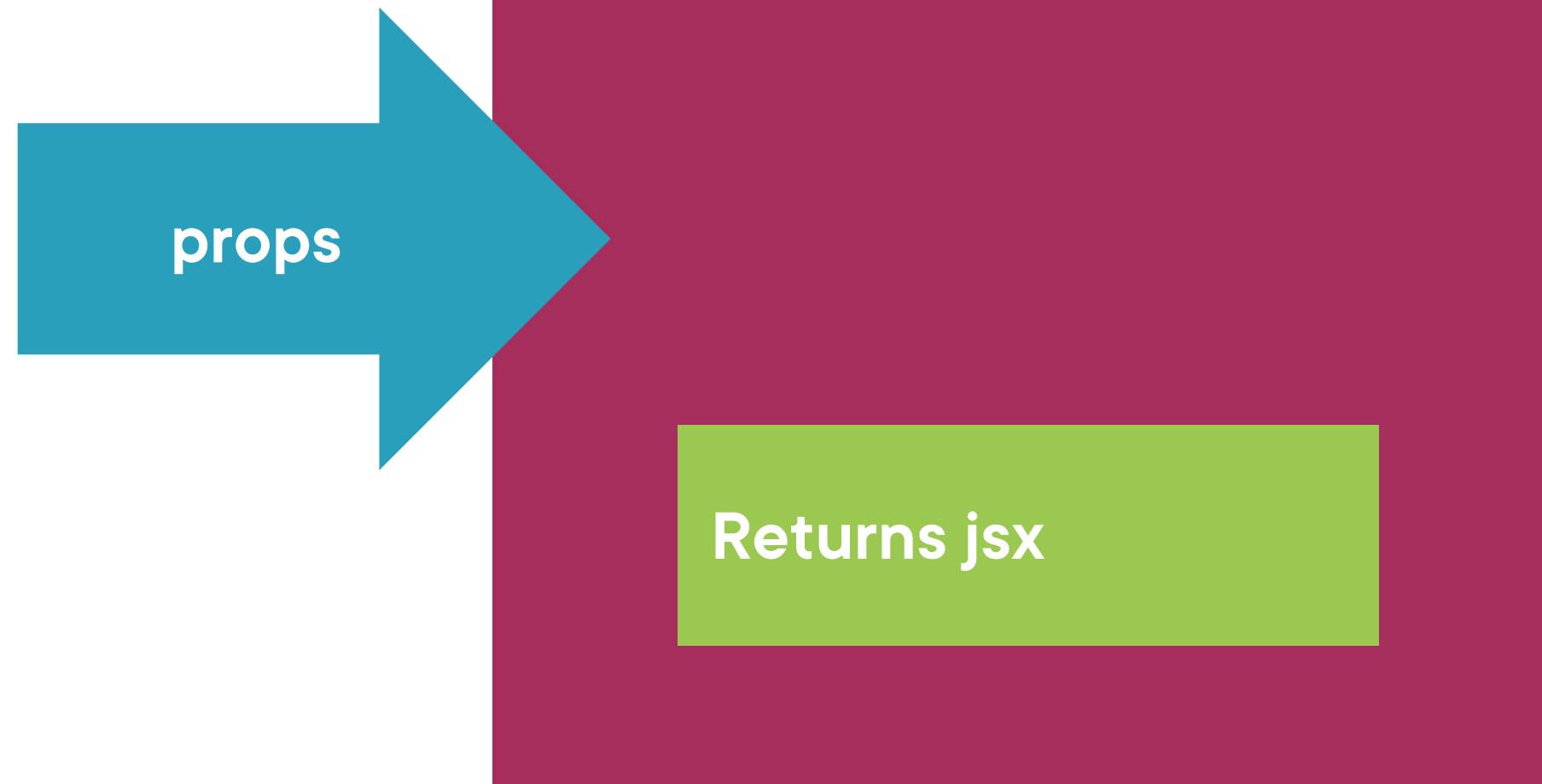


```
<Component myprop="somevalue" />
```



Props

const Component = (props) =>



<Component myprop="somevalue" />



Strict rule: Props are read-only



Props Takeaways

Passing arguments to components

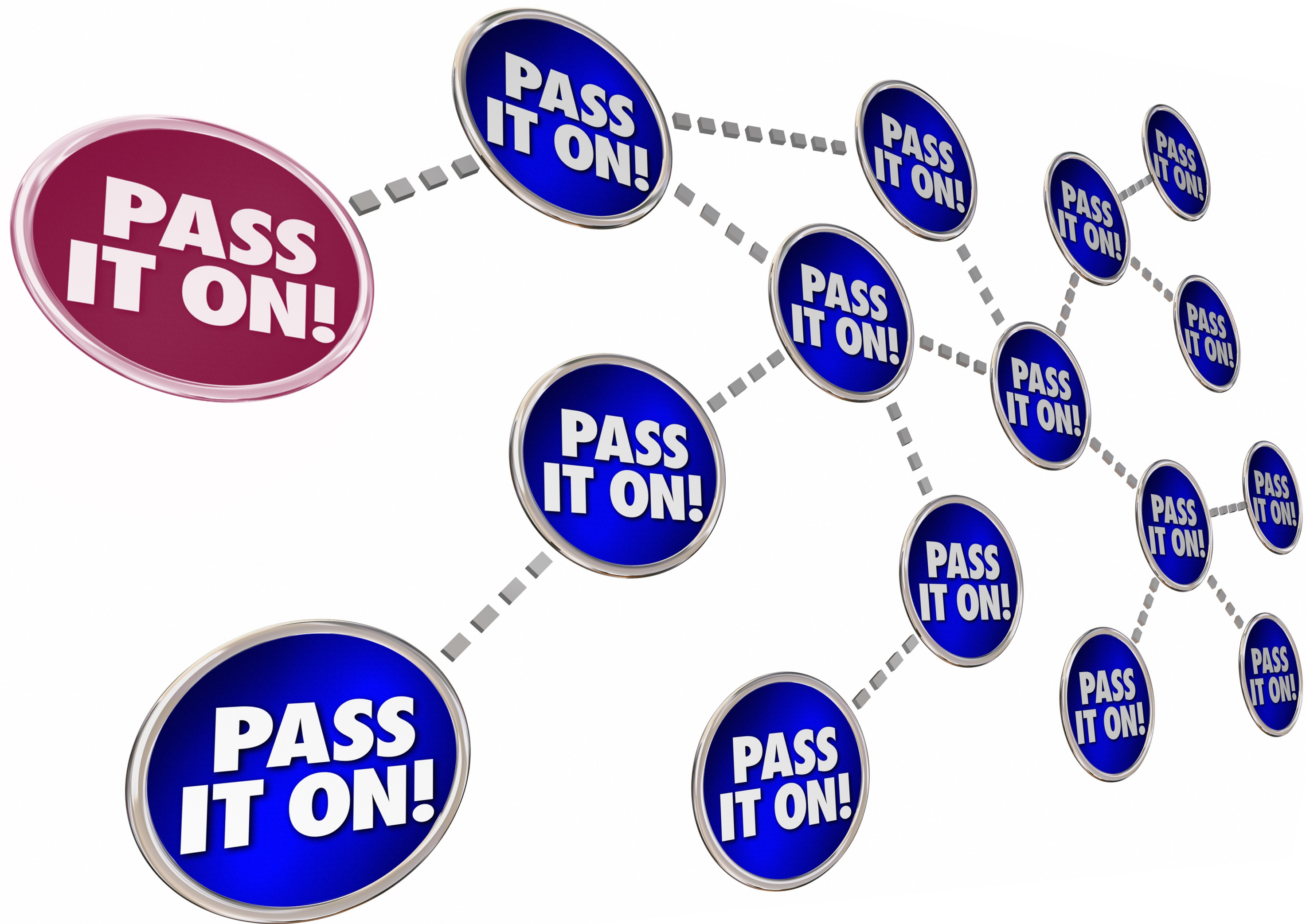
Using HTML-like attribute syntax

Receiving component accesses props object

Allow for better reusability of components

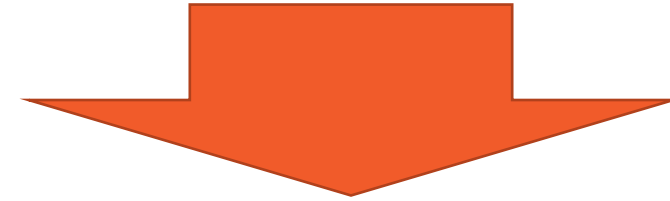




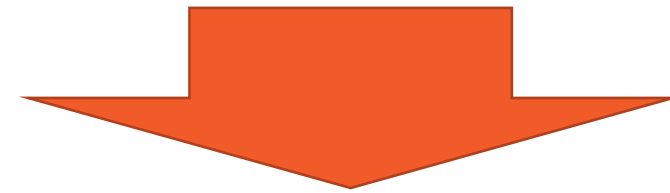


map

```
const numbers = ["one", "two", "four"];
```



```
const numbersPrefixed = numbers.map(n => "Number " + n);
```

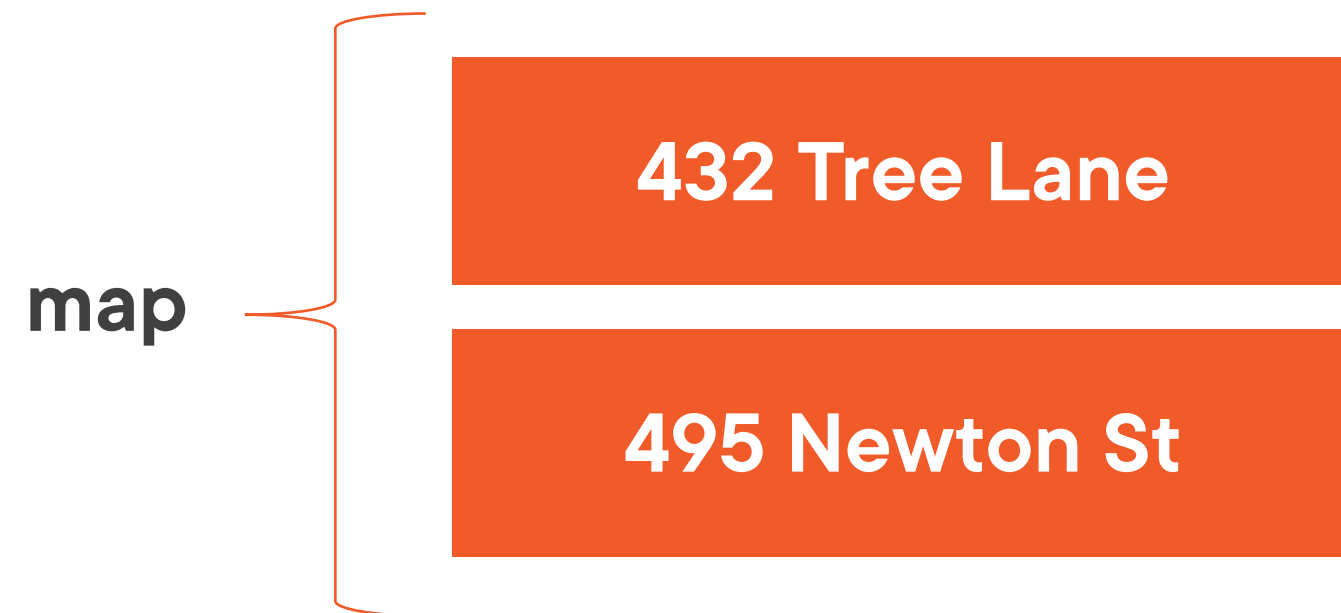


```
// ["Number one", "Number two", "Number four"]
```

The key prop

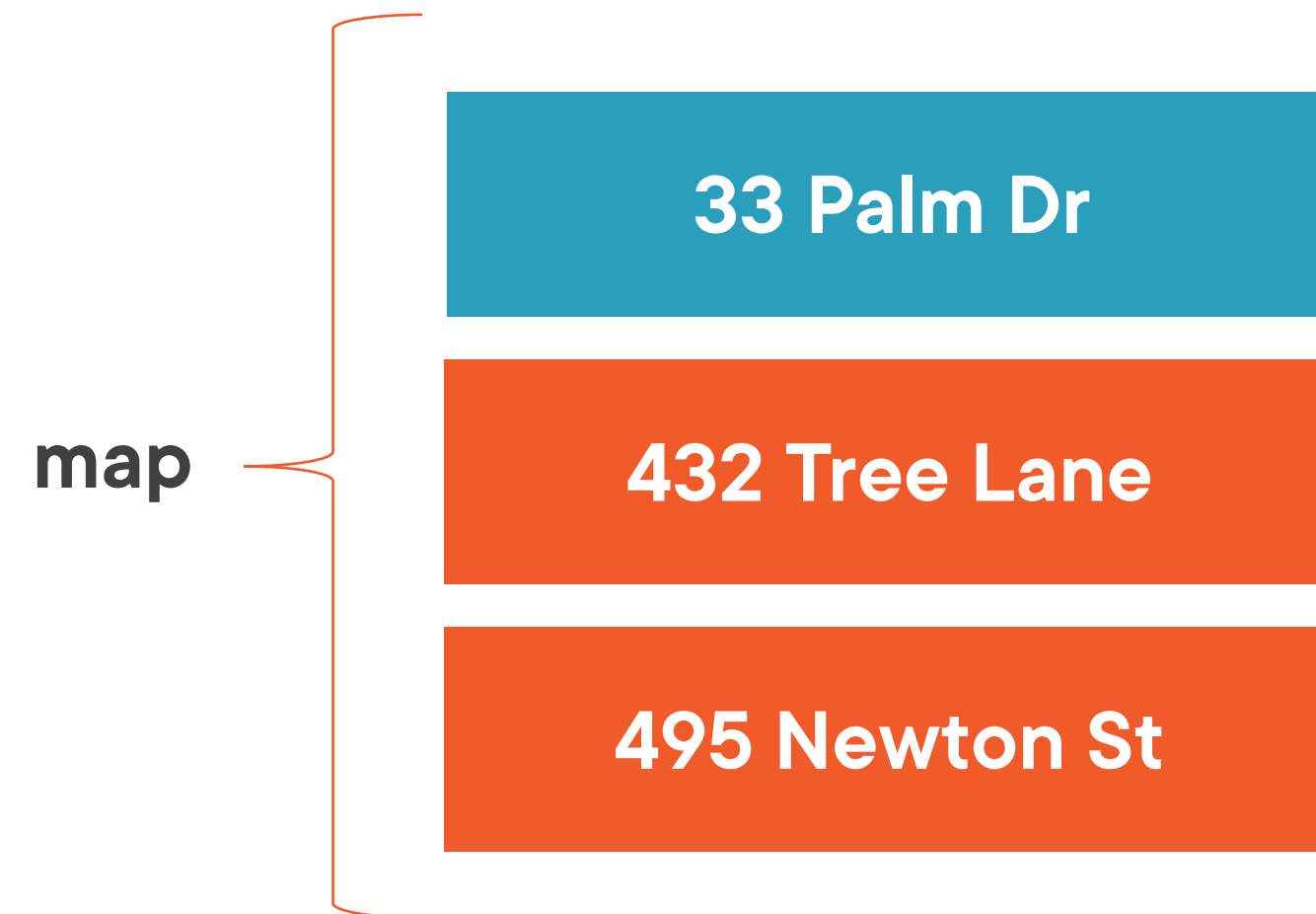
Array:

```
{ address: "432 Tree Lane" }  
{ address: "495 Newton St" }
```



Array:

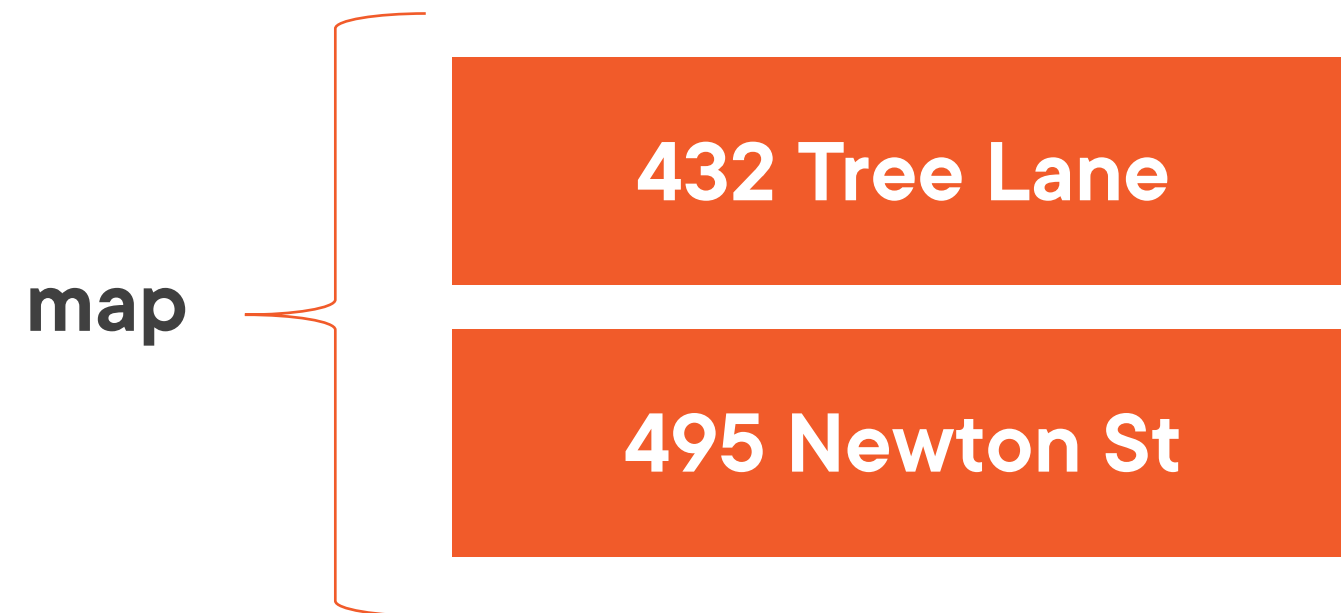
```
{ address: "33 Palm Dr" }  
{ address: "432 Tree Lane" }  
{ address: "495 Newton St" }
```



The key prop

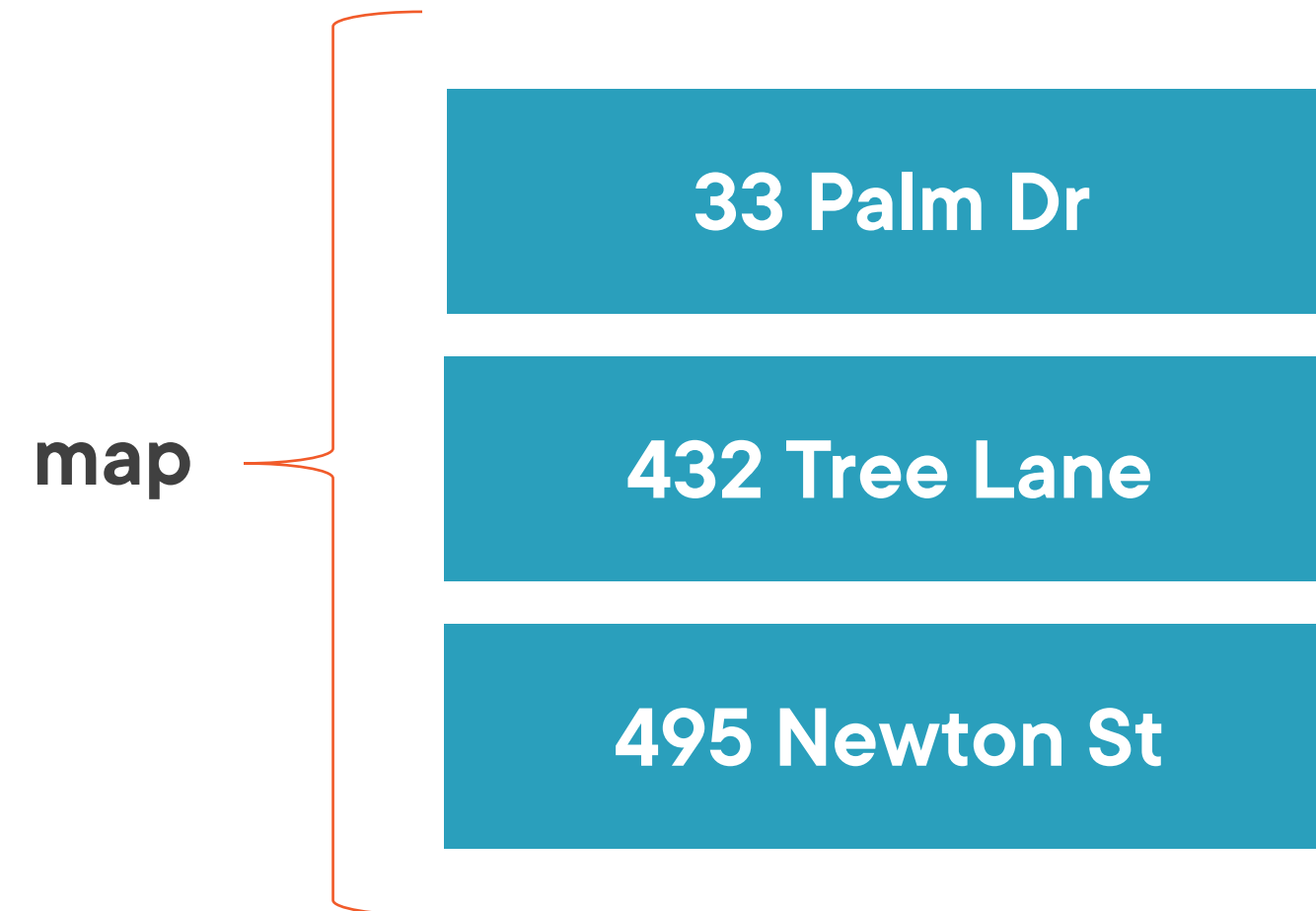
Array:

```
{ address: "432 Tree Lane" }  
{ address: "495 Newton St" }
```



Array:

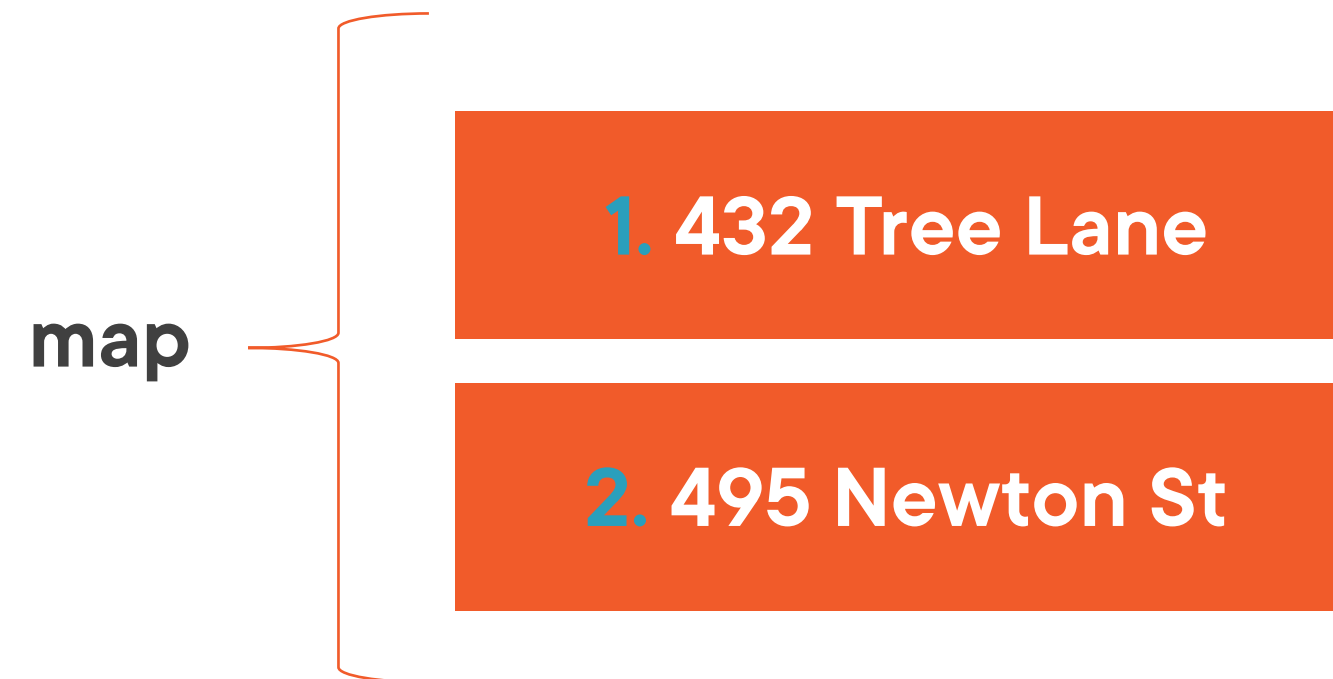
```
{ address: "33 Palm Dr" }  
{ address: "432 Tree Lane" }  
{ address: "495 Newton St" }
```



The key prop

Array:

```
{ id: 1, address: "432 Tree Lane" }  
{ id: 2, address: "495 Newton St" }
```



Choosing a Key Value

Use id property if present

**Else any combination of properties
that is unique**

Last resort: index provided by map function



Hooks

A function

Starts with "use"

Encapsulates complexity

Access React's features

Custom hooks

Imported



Rules of Hooks

Only call hooks at the top level

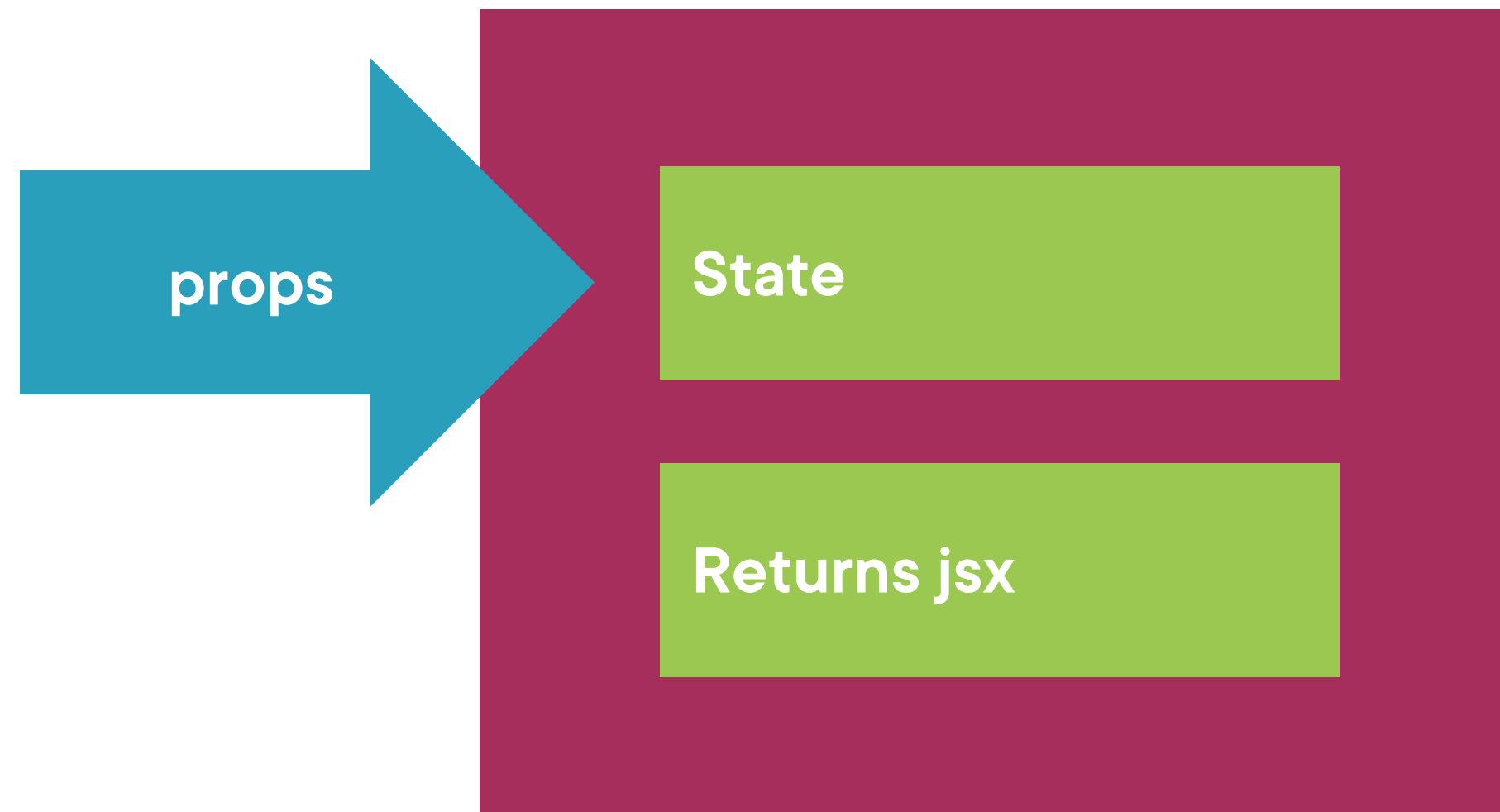
- They should always be called
- In the same order

Only call hooks in function components

- Or custom hooks

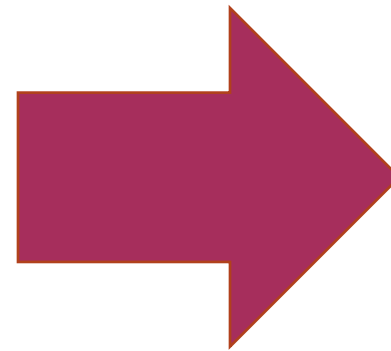


Components and State

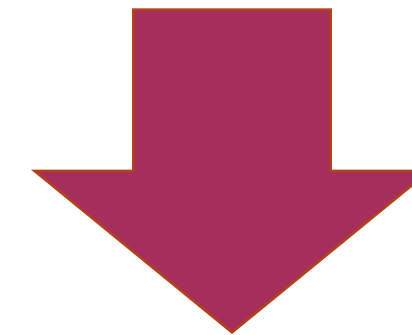


One Way Data Flow

```
[  
  {  
    id: 1,  
    address: "12 Valley of Kings, Geneva",  
    country: "Switzerland",  
    price: 900000  
  },  
  {  
    id: 2,  
    address: "89 Road of Forks, Bern",  
    country: "Switzerland",  
    price: 500000  
  },  
]
```



```
{houses.map(h => (  
  <HouseRow key={h.id} house={h} />  
))}
```



Address	Country	Asking Price
12 Valley of Kings, Geneva	Switzerland	\$900,000
89 Road of Forks, Bern	Switzerland	\$500,000



State Main Takeaways

State is internal data managed by a component

Introduced by state hook

Parameter: initial value

Returns: array with current value and function to change it (the "set function")

Change the state using the set function only



A prop value can change



React element
tr

React element
td

React element
td

React element
tr

React element
td

React element
td

React element
tr

React element
td

React element
td

React element
tr

React element
td

React element
td

React element
tr

React element
td

React element
td



React element
tr

React element
td

React element
td

React element
tr

React element
td

React element
td

React element
tr

React element
td

React element
td

React element
tr

React element
td

React element
td

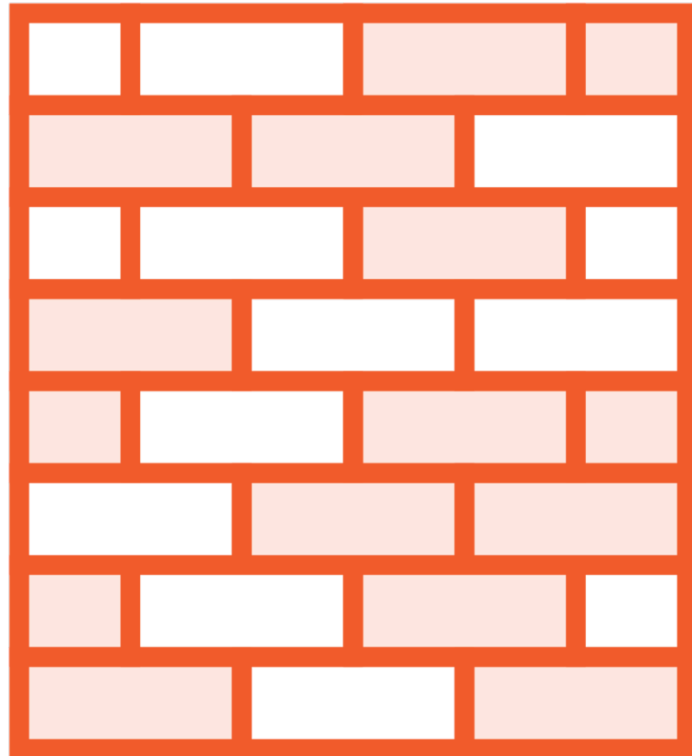
React element
tr

React element
td

React element
td



Core React Features



Structure with components
Reusability
Have state



UIs declared in JavaScript
Rendered output changes
when state is updated



Efficiency with reconciliation
Only updates the parts
of the UI that changed



Next up:

Component Rendering and Side Effects

