


Alternative way to access props

Wednesday, March 23, 2022 3:41 PM

- We may pass multiple prop value to a component
- All those prop values are collected as a single object

```
<Cell id={2} onCellClick={onCellClick} value="0" />
```

- All the props are passed as a single object
- Now we Cell Component as Below



```
const Cell=(props)=>{  
  let value = props.value || '-'; //if props.value === falsy use '-'  
  let style={  
    color: props.value ? "black" : "transparent"  
  };  
  
  return <button  
    style={style}  
    onClick={ ()=> props.onCellClick(props.id)}  
    className="cell">{value}</button>;  
}
```

```
{  
  value:"0",  
  id:2,  
  onCellClick:function
```

- The Problem is
 - you need to use **props**. Everywhere
 - You need to create local variable if you need to work on it.

We can use ES2015 parameter de-structure syntax

```
const Cell=({id, value, onCellClick})=>{  
  value = value || '-'; //if props.value === falsy use '-'  
  let style={  
    color: value==='-' ? "transparent" : null  
  };  
  
  return <button  
    style={style}  
    onClick={ ()=> onCellClick(id)}  
    className="cell">{value}</button>;  
}
```

- Whatever individual props you are expecting can be supplied this way
- We don't have to use the **props**. anymore

- We may also specify default values