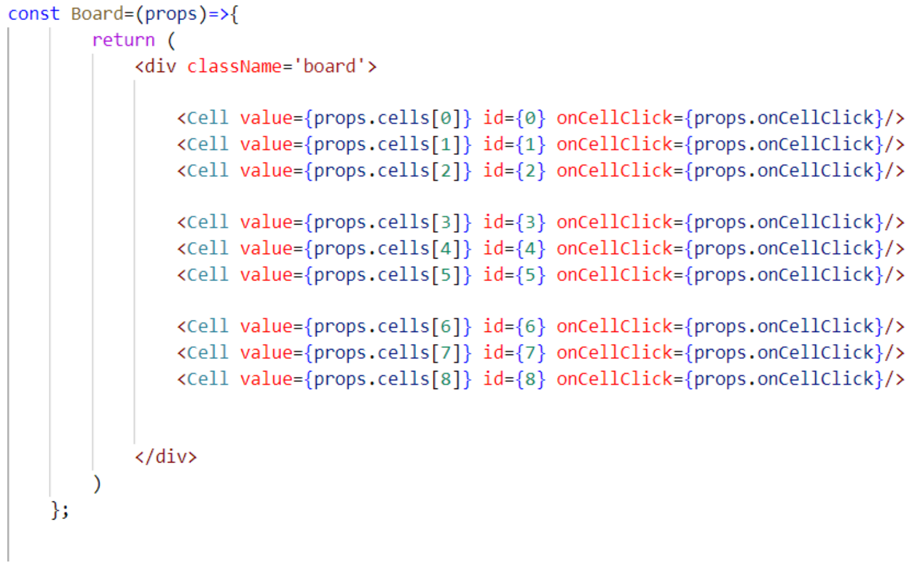
Tic Tac Toe Cell generation

Wednesday, March 23, 2022 5:25 PM

Consider our Board Logic

- We generate 9 cells



```
const Board=(props)=>{
        return (
            <div className='board'>

                <Cell value={props.cells[0]} id={0} onCellClick={props.onCellClick}/>
                <Cell value={props.cells[1]} id={1} onCellClick={props.onCellClick}/>
                <Cell value={props.cells[2]} id={2} onCellClick={props.onCellClick}/>

                <Cell value={props.cells[3]} id={3} onCellClick={props.onCellClick}/>
                <Cell value={props.cells[4]} id={4} onCellClick={props.onCellClick}/>
                <Cell value={props.cells[5]} id={5} onCellClick={props.onCellClick}/>

                <Cell value={props.cells[6]} id={6} onCellClick={props.onCellClick}/>
                <Cell value={props.cells[7]} id={7} onCellClick={props.onCellClick}/>
                <Cell value={props.cells[8]} id={8} onCellClick={props.onCellClick}/>


            </div>
        )
    };
```

- This is a lot of redundant logic here
- What if we generate a check board (64 cells)
- Or large Minesweeper (200cells)

We can use map() to generate a Cell for each item in props.cells array