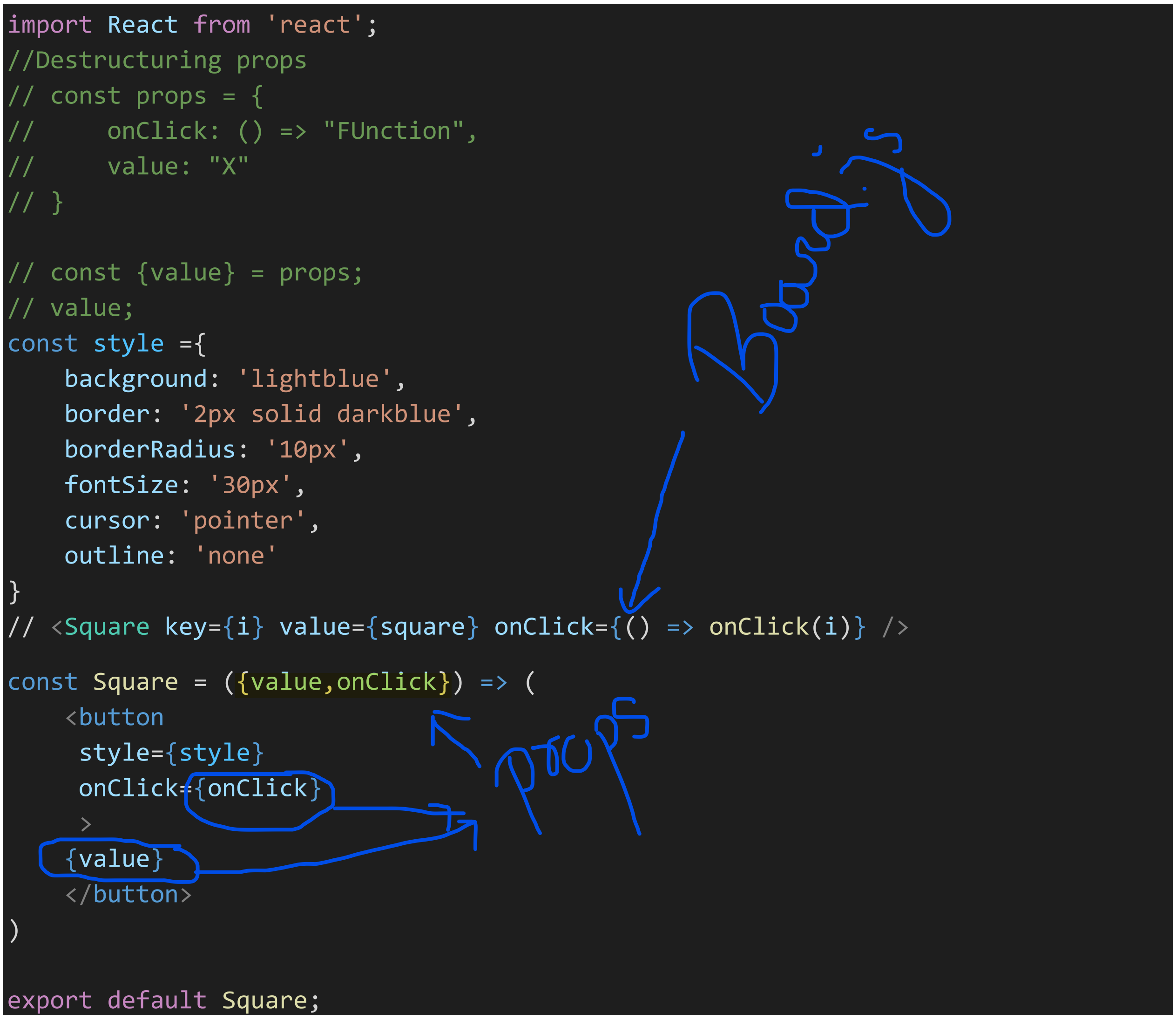
Folder Structure:

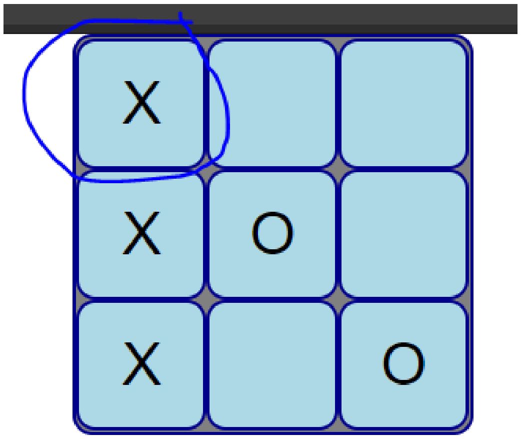
Create components folder in src folder.

Inside components folder, we will create these JS files.

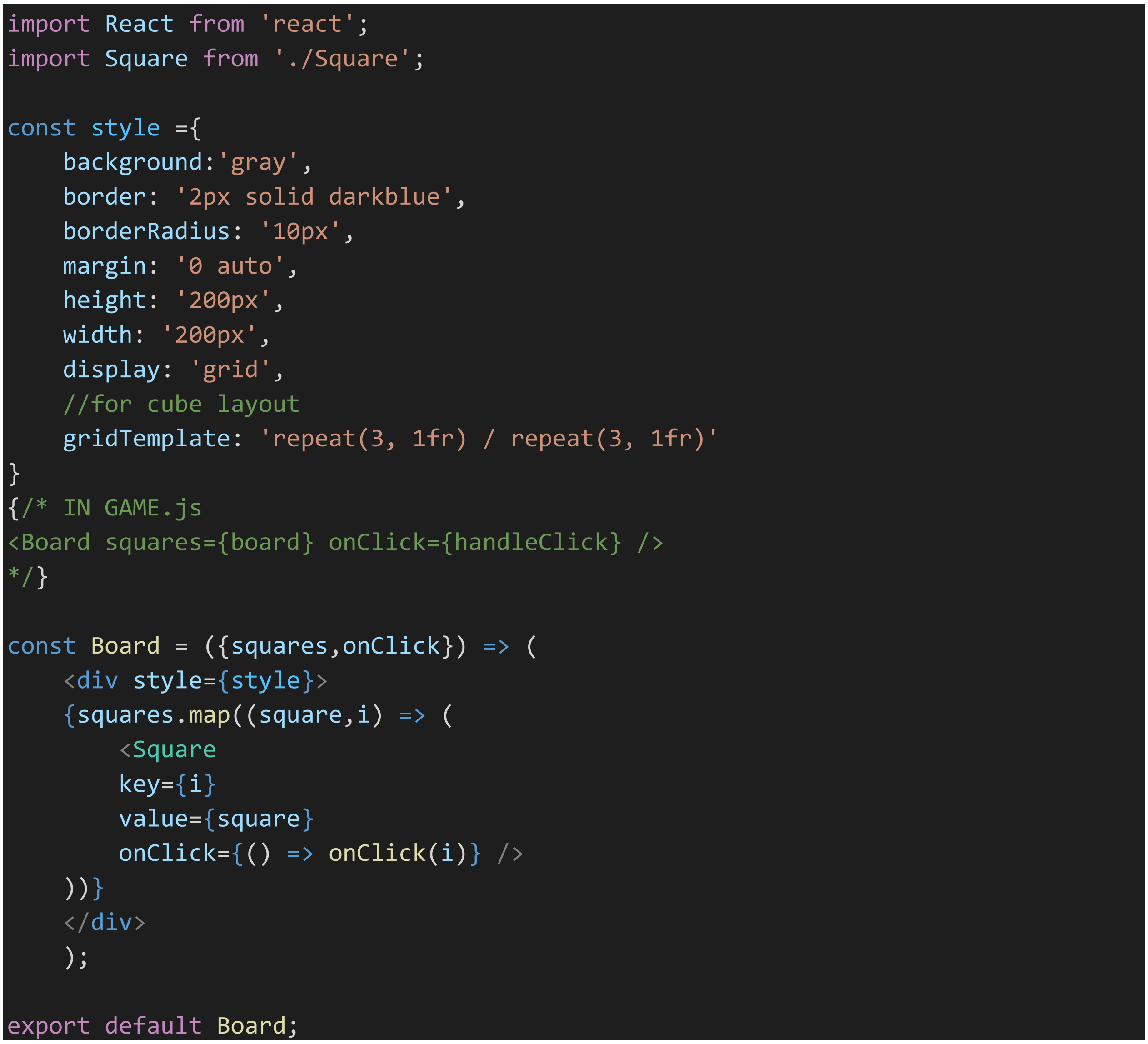
Square.js



```
import React from 'react';
//Destructuring props
// const props = {
//     onClick: () => "FUnction",
//     value: "X"
// }

// const {value} = props;
// value;
const style ={
    background: 'lightblue',
    border: '2px solid darkblue',
    borderRadius: '10px',
    fontSize: '30px',
    cursor: 'pointer',
    outline: 'none'
}
// <Square key={i} value={square} onClick={() => onClick(i)} />

const Square = ({value,onClick}) => (
    <button
     style={style}
     onClick={onClick}
     >
    {value}
    </button>
)

export default Square;
```



Board.js



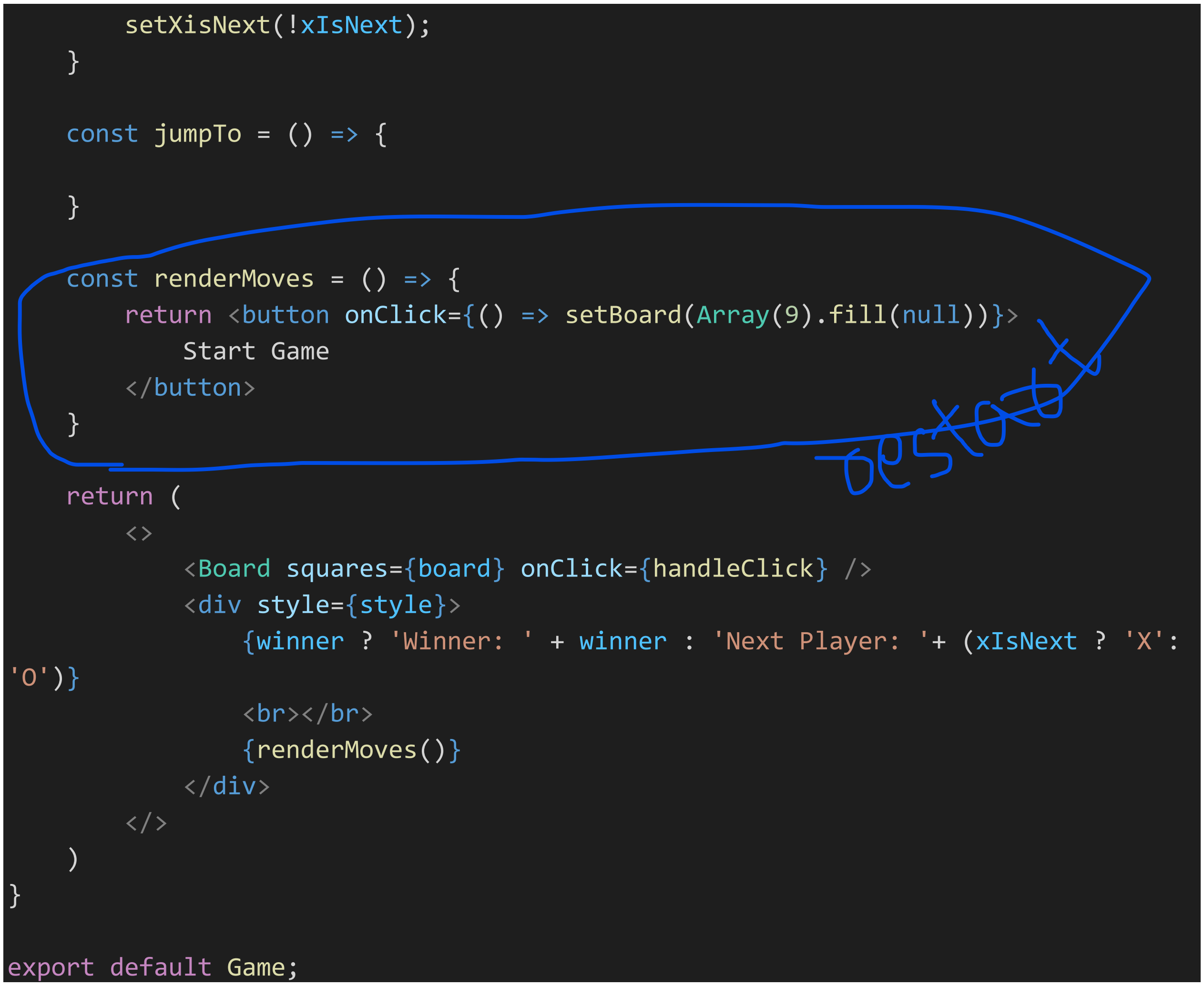
```
import React from 'react';
import Square from './Square';

const style ={
    background:'gray',
    border: '2px solid darkblue',
    borderRadius: '10px',
    margin: '0 auto',
    height: '200px',
    width: '200px',
    display: 'grid',
    //for cube layout
    gridTemplate: 'repeat(3, 1fr) / repeat(3, 1fr)'
}
{/* IN GAME.js
<Board squares={board} onClick={handleClick} />
*/}

const Board = ({squares,onClick}) => (
    <div style={style}>
    {squares.map((square,i) => (
        <Square
        key={i}
        value={square}
        onClick={() => onClick(i)} />
    ))}
    </div>
    );

export default Board;
```

.map le chai iterative ma square haru banauxa

```
const Board = (squares,onClick) => (
    <div style={style}>
        <Square value={1} onClick={() => onClick("{onClick}")} />
        <Square value={1} onClick={() => onClick("{onClick}")} />
        <Square value={1} onClick={() => onClick("{onClick}")} />
        <Square value={1} onClick={() => onClick("{onClick}")} />
        <Square value={1} onClick={() => onClick("{onClick}")} />
        <Square value={1} onClick={() => onClick("{onClick}")} />
        <Square value={1} onClick={() => onClick("{onClick}")} />
        <Square value={1} onClick={() => onClick("{onClick}")} />
        <Square value={1} onClick={() => onClick("{onClick}")} />
    </div>
    );
```

Game.js

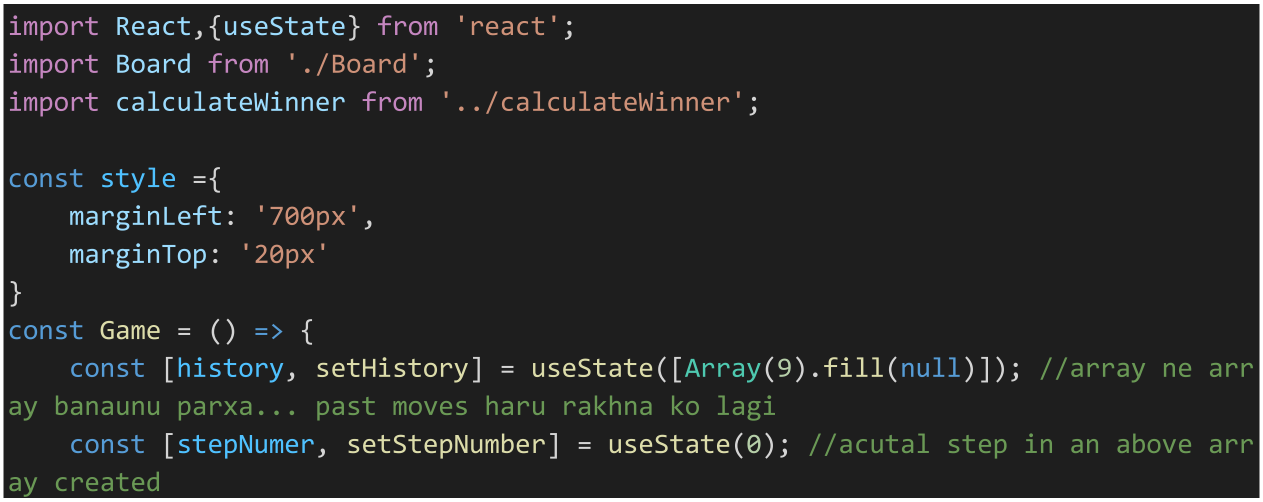
```
import React,{useState} from 'react';
import Board from './Board';
import calculateWinner from '../calculateWinner';

const style ={
    marginLeft: '700px',
    marginTop: '20px'
}
const Game = () => {
    const [board, setBoard] = useState(Array(9).fill(null));
    //destructuring above state
    // const board = useState("state");
    // board[0]
    // board[1]
    const [xIsNext, setXisNext] = useState(true);
    const winner = calculateWinner(board);

    const handleClick = (i) =>{
        //copying board state. immutable
        const boardCopy = [...board];
        //if user click on occupied square or if game is won, return
        if( winner || boardCopy[i] ) return; //yedi winner vayeskayo ya lekhisake
ko square ma click garyo vane nothing return
        //put an x or an o in the clicked square
        boardCopy[i] = xIsNext ? 'X' : 'O'; //i = kun square ho vanera
        //yedi square[1] ma click garyo vane X dine state true cha vane natra O
        setBoard(boardCopy);
```



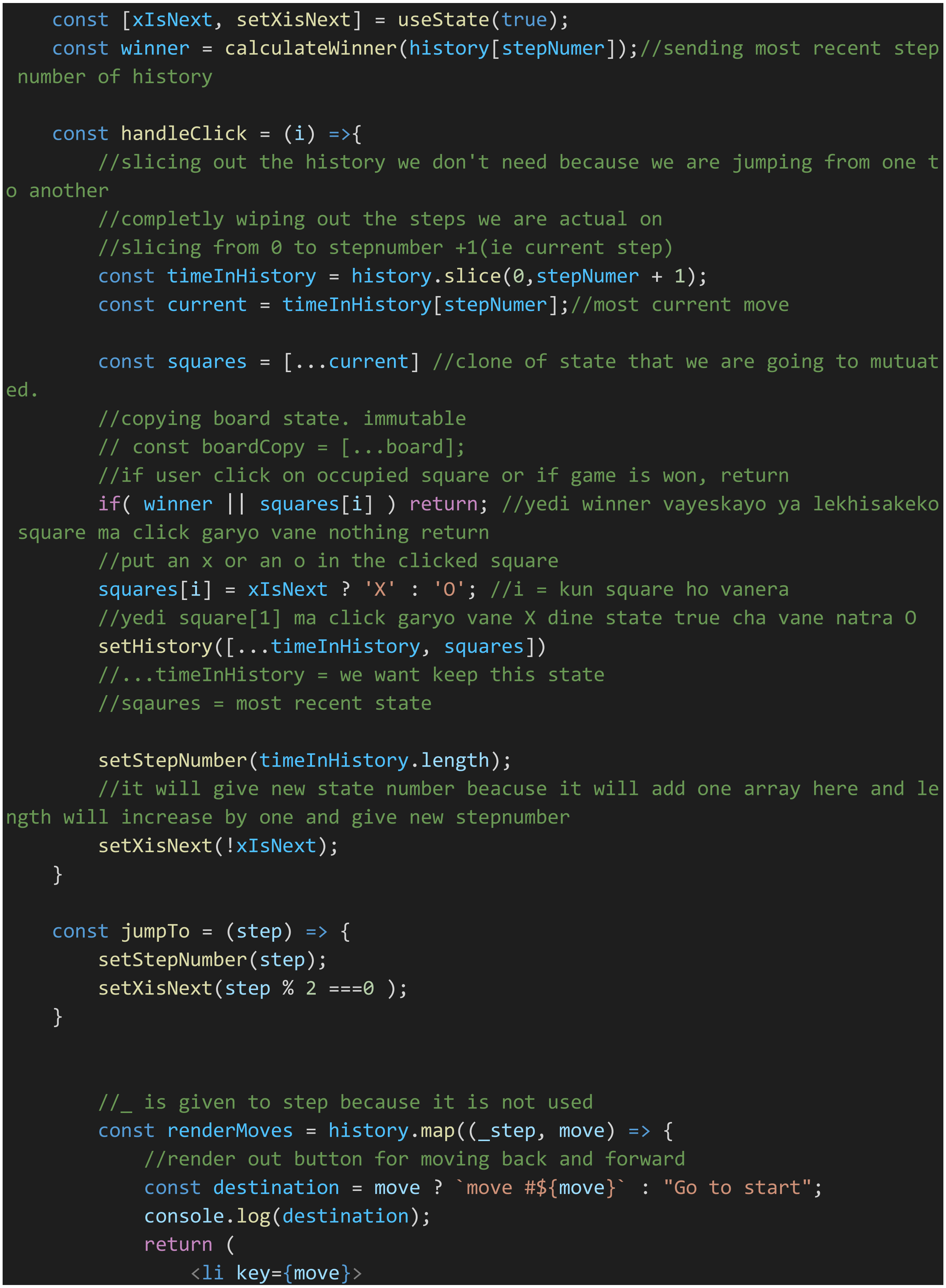
```
        setXisNext(!xIsNext);
    }

    const jumpTo = () => {

    }

    const renderMoves = () => {
        return <button onClick={() => setBoard(Array(9).fill(null))}>
            Start Game
        </button>
    }

    return (
        <>
            <Board squares={board} onClick={handleClick} />
            <div style={style}>
                {winner ? 'Winner: ' + winner : 'Next Player: '+ (xIsNext ? 'X':
'O')}
                <br></br>
                {renderMoves()}
            </div>
        </>
    )
}

export default Game;
```

FINAL

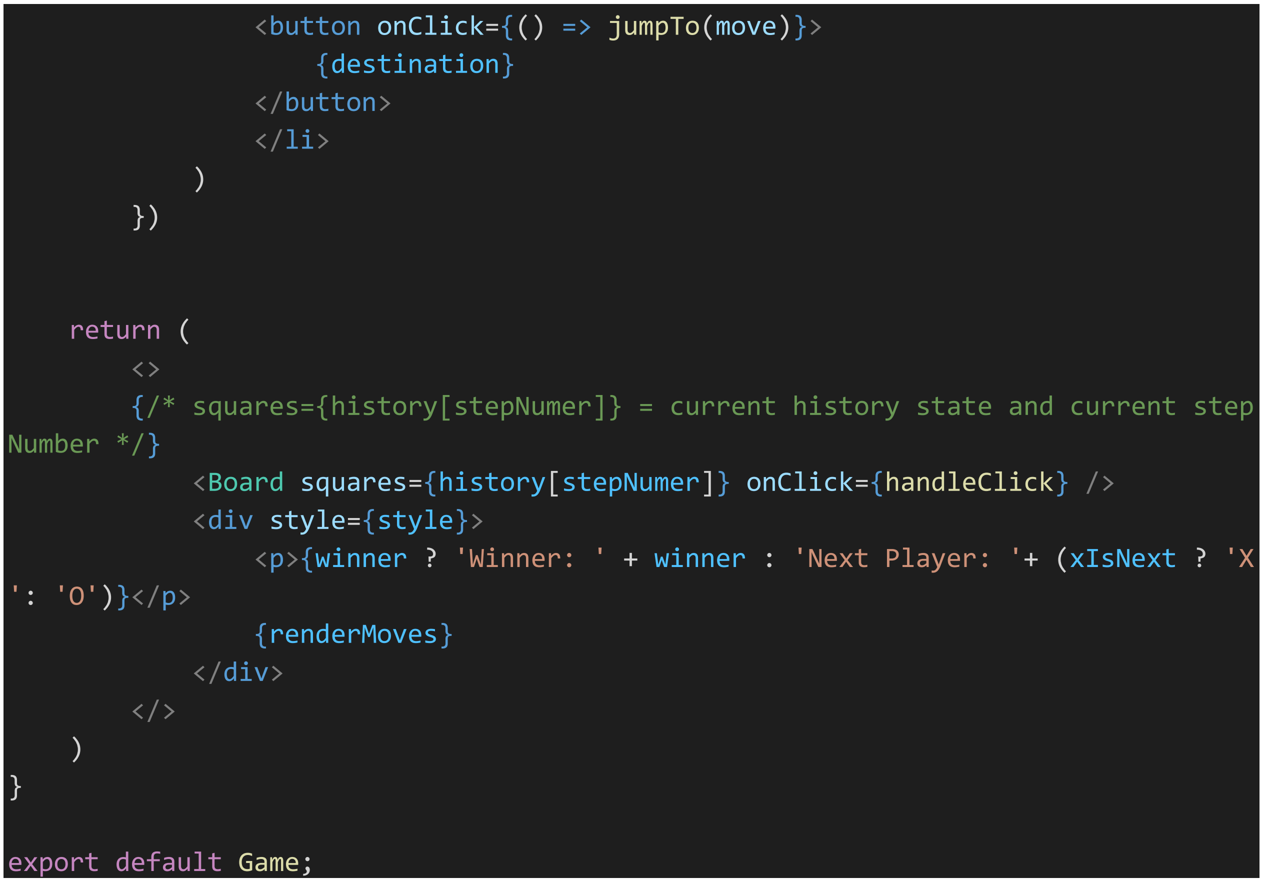
Game.js



```
import React,{useState} from 'react';
import Board from './Board';
import calculateWinner from '../calculateWinner';

const style ={
    marginLeft: '700px',
    marginTop: '20px'
}
const Game = () => {
    const [history, setHistory] = useState([Array(9).fill(null)]); //array ne arr
ay banaunu parxa... past moves haru rakhna ko lagi
    const [stepNumer, setStepNumber] = useState(0); //acutal step in an above arr
ay created
```



```
  const [xIsNext, setXisNext] = useState(true);
  const winner = calculateWinner(history[stepNumer]);//sending most recent step
 number of history

  const handleClick = (i) =>{
    //slicing out the history we don't need because we are jumping from one t
o another
    //completly wiping out the steps we are actual on
    //slicing from 0 to stepnumber +1(ie current step)
    const timeInHistory = history.slice(0,stepNumer + 1);
    const current = timeInHistory[stepNumer];//most current move

    const squares = [...current] //clone of state that we are going to mutuat
ed.
    //copying board state. immutable
    // const boardCopy = [...board];
    //if user click on occupied square or if game is won, return
    if( winner || squares[i] ) return; //yedi winner vayeskayo ya lekhisakeko
 square ma click garyo vane nothing return
    //put an x or an o in the clicked square
    squares[i] = xIsNext ? 'X' : 'O'; //i = kun square ho vanera
    //yedi square[1] ma click garyo vane X dine state true cha vane natra O
    setHistory([...timeInHistory, squares])
    //...timeInHistory = we want keep this state
    //sqaures = most recent state

    setStepNumber(timeInHistory.length);
    //it will give new state number beacuse it will add one array here and le
ngth will increase by one and give new stepnumber
    setXisNext(!xIsNext);
  }

  const jumpTo = (step) => {
    setStepNumber(step);
    setXisNext(step % 2 ===0 );
  }


    //_ is given to step because it is not used
    const renderMoves = history.map((_step, move) => {
      //render out button for moving back and forward
      const destination = move ? `move #${move}` : "Go to start";
      console.log(destination);
      return (
        <li key={move}>
```



```
                <button onClick={() => jumpTo(move)}>
                    {destination}
                </button>
                </li>
            )
        })


    return (
        <>
        {/* squares={history[stepNumer]} = current history state and current step
Number */}
            <Board squares={history[stepNumer]} onClick={handleClick} />
            <div style={style}>
                <p>{winner ? 'Winner: ' + winner : 'Next Player: '+ (xIsNext ? 'X
': 'O')}</p>
                {renderMoves}
            </div>
        </>
    )
}

export default Game;
```