*1. React – Mount/Unmount*

In the Tic-tac-toe video, you saw how to mount/unmount complete rows of squares using a button. In this activity, you’ll dig deeper into the mount/unmount component lifecycle events.

A React component’s lifecycle contains distinct phases for creation and deletion. These are called:

* *mounting = creation*
* *unmounting = deletion*

When the [Show/Hide Row] button is clicked, you’re telling React to create or delete all of the squares entirely from the DOM.

Just like you saw in the video lesson, there’s a button that will mount/unmount the entire row. In this assignment, you need to move the unmounting logic from the board component to the square component, so that when you click on a square, that square will be the only one that unmounts.

The starter code for this activity is in the lifecycle.jsx file.

**In this activity, your task is to have each individual square unmount whenever when it is clicked**

Once you’re done, the game board should function as follows:

A picture containing text, monitor

Description automatically generated

**Where did the square go?**

Once a square is unmounted (deleted), clicking on the square again won’t bring it back to the board. But why? When you unmount the square, you told React to delete the square component from the DOM. Since you moved the mount/unmount logic out of the game board and into each individual square component, the HTML was deleted from the DOM when you unmounted (deleted) the square.

In order to bring a square back to the board, you need to refresh the entire Web Browser so that React will mount (create) them again.

**In an upcoming activity, instead of mounting/unmounting to make changes, you’ll be re-rendering components**

*Notes:*

* *The starter code includes helpful comments to guide you through this activity*

*Hint:*

* *Make sure you move ALL of the code related to*mounted*from the board component, paying particular attention to the return function. Otherwise, there will be errors.*