Hints 2: Tracking Grid Clicks

To track and store the grid clicks, you can perform the following steps in the JavaScript file:

1. Declare a variable named `currentPlayerSymbol` and set it to "x".
2. Declare a variable named `squareValues` and initialize it to an array with nine empty string entries.
3. Add an event listener to the `window` object for the `DOMContentLoaded` event.
4. In the event handler for the `DOMContentLoaded` event, add a click event handler to the tic-tac-toe grid `<div>`.
5. In the click event handler, inspect the target of the event. Ignore the event if its `id` does not begin with "square-".
6. If the event target's `id` attribute does begin with "square-", then parse the number after "square-" using the Number.parseInt method.
7. If the value in the `squareValues` array for the index of the parsed number is not an empty string, then ignore the event.
8. There must not be a play in that square, so:

   - Programmatically create an `img` element, set its source to appropriate value from the requirements by using the value in `currentPlayerSymbol`, and append the `img` element to the event target.
   - Store the value of `currentPlayerSymbol` in the corresponding slot in the `squareValues` array.
   - If the `currentPlayerSymbol` is "x", then set it to "o". Otherwise, set `currentPlayerSymbol` to "x".

Why do these hints suggest creating an array in which to track the symbols? In the world of good programming, the *state* of the application is kept separate from the *visualization* of the program's state. What that practically means is that the Tic Tac Toe game really only cares about the "x" values and the "o" values, not the fact that there are fancy SVG images in the UI. For it to figure out the winner or tie, it cares about the *data* that describes the game board at any given point, not the HTML elements that the players

see. If the program knows the state of the board at any given point, then it can make those determinations as well as restoring the game board from that point in the future.

When you finish, you should have something similar to the following.