

Breaking down the project

+ psuedocode

Keywords:

- Hosted game
- Switch between players
- Display winner
- JS DOM manipulation
- HTML, CSS, JS files separate

HTML Breakdown:

- Grids - divs
- Display 'Player 1' and 'Player 2'

```
<body>
  section.grid-container>div*9
</body>
```

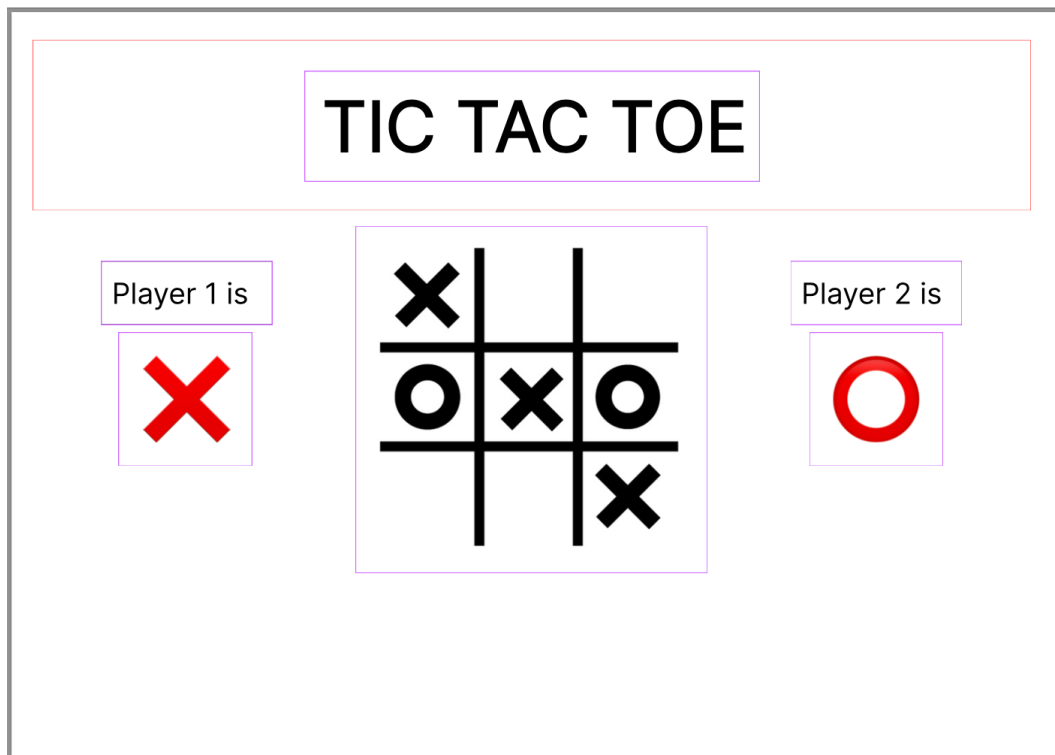
CSS Breakdown:

- Style the divs to show grid
 - Display appropriate border
 - Can you use `display: grid` for the grid? And colour the gap in to show grid lines?

```
.grid-container {
  display as grid
  3 columns 3 row
  gap-color black???
}
```

- Spacing between grid and winner text

Wireframe:



JS Breakdown:

- Store each player's score
 - The value will then be displayed on HTML
 - This will act as a scoreboard

```
var player1Score
var player2Score
var newParagraph1
var newParagraph2

if player1 wins > player1Score++
if player2 wins > player2Score++

newParagraph1 = player1Score
newParagraph2 = player2Score

body.appendChild(newParagraph1) // add to scoreboard
body.appendChild(newParagraph2) // add to scoreboard
```

- Add event to user's click -
 - link click to each square of the grid via DOM
 - Each alternating click is 'O' and other clicks are 'X'

e.g click 1 = X, click 2 = O, click 3 = X, click 4 = other

```
var allDivContainer = divContainer.DOM

// first click = always player1, second click = player2

allDivContainer.click {
  gridSquare = event.target

  if player1Click is div# {
    print 'X'}
  }

  if player2Click is div# {
    print 'O'
  }

  // Breaking 'player click' down further:
  player1Click = every odd click
  player2Click = every even click
  // Can you target each click like that?
```

- Switch between players:
 - Display text to show which player's turn, depending on which mark was left on previous click - **if statement**

e.g if 'X' is player 1 and last click was 'X', then display player 2's turn (next click is 'O')

```
// Following from above psuedocode:

if player1Click is div# {
  print 'X'}
  display "player 2's turn"
}

if player2Click is div# {
  print 'O'
  display "player 1's turn"
}

// Breaking 'display' down further:
HTML:
```

```

Create <p> tag:
"It's player 1's turn"
"It's player 2's turn"

CSS:
.p {
  display: none;
}

.show-turn {
  display: block;
}

JS:
Toggle 'show-turn' class to display whose turn it is.

```

- Display winner:
 - When 3 in a row, show a message (new modal, new text)
 - Append to HTML

```

// 8 possible ways to win for each player in this specific TTT game:
XXX   ...   ...   X..   .X.   ..X   X..   ..X
...   XXX   ...   X..   .X.   ..X   .X.   .X.
...   ...   XXX   X..   .X.   ..X   ..X   X..

if any combo is met, display a winner message

// Breaking down further:
if 'X' is in div 1, 2, 3 {
  alert('Player 1 won!')
}

// or
var newH2
var winner = empty var

if 'X' is in div 1, 2, 3 {
  winner = player that won
  newH2 = winner
  body.appendChild.newH2
}

```