game.js

```
var grid = [
    // each item is a row
    ["none", "none", "none"],
    ["none", "none", "none"],
    ["none", "none", "none"]];
var currentPlayer, gameType, allowed, winLocation;
/*  gameType values:
        0: 2-Player
        1: 1P Easy
        2: 1P Hard  */
var xWins = 0;
var oWins = 0;

function resetGame() {
    grid = [
        ["none", "none", "none"],
        ["none", "none", "none"],
        ["none", "none", "none"]];
    for (a in grid) { //reset table on the page
        for (b in grid[a]) {
            document.getElementById(b.toString() + a.toString()).src = "none.png";
        }
    }
    document.getElementById('message').innerHTML = " ";

}
function newGame() {
    resetGame()
    allowed = true;
    if (gameType == 0) {
        document.getElementById('message').innerHTML = currentPlayer + "'s turn...";
    } else if (currentPlayer == 'x') {
        document.getElementById('message').innerHTML = "Your turn.";
    } else {
        easyGame()
    }
}
function winner(player) { //function checks if the given player is the winner
    for (i in grid) {
        //checks rows
        if (grid[i][0] == player && grid[i][1] == player && grid[i][2] == player) {
            return true;
        }
        //checks columns
        if (grid[0][i] == player && grid[1][i] == player && grid[2][i] == player) {
            return true;
        }
    }
    //checks diagonals
    if (grid[0][0] == player && grid[1][1] == player && grid[2][2] == player) {
        return true;
    } else if (grid[0][2] == player && grid[1][1] == player && grid[2][0] == player) {
        return true;
    }
    // If player hasn't won..
    for (a in grid) { //if board isn't full, return false (game isn't over)
        if (grid[a].includes('none')) {
            return false;
        }
    }
    //undefined means tie
    return undefined;
}
function updateScoreboard() {
    document.getElementById('xWins').innerHTML = xWins;
    document.getElementById('oWins').innerHTML = oWins;
```

```
}

function startGame(type) {
    resetGame()
    xWins = 0;
    oWins = 0;
    updateScoreboard()
    //choose starting player
    if (Math.random() >= .5) {
            currentPlayer = "o";
    } else {
        currentPlayer = "x";
    }
    gameType = type;
    allowed = true;
    if (type == 0) {
        document.getElementById('message').innerHTML = currentPlayer.toUpperCase() +
        "'s turn.";
    } else if (currentPlayer == 'x') {
        document.getElementById('message').innerHTML = "Your turn.";
    } else {
        easyGame()
    }

}

/* HUMAN'S TURN, Main Game */

function playTurn(trow, tcol, player) {
    //only lets player take turn if it's their turn
    if ((currentPlayer == 'x' || gameType == 0) && grid[trow][tcol] == "none" &&
    allowed) {
        grid[trow][tcol] = player;
        document.getElementById(trow.toString() + tcol.toString()).src = player + ".png";
        //switch players
        if (player == 'x') {
            currentPlayer = 'o';
        } else {
            currentPlayer = 'x';
        }
        //end game if player wins
        if (winner(player)) {
            if (gameType == 0) {
                document.getElementById('message').innerHTML = player + " wins!<br
                /><button onclick='newGame()'>New Game</button>";
            } else {
                document.getElementById('message').innerHTML = "You win!<br /><button
                onclick='newGame()'>New Game</button>";
            }
            allowed = false;
            //increase score of winner
            if (player == 'o') {
                oWins++;
            } else {
                xWins++;
            }
            updateScoreboard()
        } else if (winner(player) == undefined) {
            //tie
            document.getElementById('message').innerHTML = "Tie!<br /><button
            onclick='newGame()'>New Game</button>";
            allowed = false;
        } else if (gameType == 0) {
            document.getElementById('message').innerHTML = currentPlayer.toUpperCase()
            + "'s turn.";
        }
```

```
        //both only run if in the correct game type
        easyGame()
    }
}

/* "AI"!
   (2 player mode needs no function since playTurn only runs for X or 2 player)
    uses algorithm, not real AI
    The name easyGame() is a remnant from the original plan to make separate easy and
    hard AI modes
    however, the easy AI was TOO easy, so now it's just 1, medium difficulty AI. */

function nextWin(player) { //makes it possible to block or win when opportunity exists
    for (i in grid) {
        //top or left is blank
        if (grid[i][0] == "none" && grid[i][1] == player && grid[i][2] == player) {
            winLocation = [i, 0]
            return true;
        }
        if (grid[0][i] == "none" && grid[1][i] == player && grid[2][i] == player) {
            winLocation = [0, i]
            return true;
        }
        //middle blank
        if (grid[i][0] == player && grid[i][1] == "none" && grid[i][2] == player) {
            winLocation = [i, 1]
            return true;
        }
        if (grid[0][i] == player && grid[1][i] == "none" && grid[2][i] == player) {
            winLocation = [1, i]
            return true;
        }
        //bottom or right blank
        if (grid[i][0] == player && grid[i][1] == player && grid[i][2] == "none") {
            winLocation = [i, 2]
            return true;
        }
        if (grid[0][i] == player && grid[1][i] == player && grid[2][i] == "none") {
            winLocation = [2, i]
            return true;
        }
    }
    //DIAGONALS
    //top is blank
    if (grid[0][0] == "none" && grid[1][1] == player && grid[2][2] == player) {
        winLocation = [0, 0]
        return true;
    } else if (grid[0][2] == "none" && grid[1][1] == player && grid[2][0] == player) {
        winLocation = [0, 2]
        return true;
    }
    // center is blank
    if (grid[0][0] == player && grid[1][1] == "none" && grid[2][2] == player) {
        winLocation = [1, 1]
        return true;
    } else if (grid[0][2] == player && grid[1][1] == "none" && grid[2][0] == player) {
        winLocation = [1, 1]
        return true;
    }
    //bottom is blank
    if (grid[0][0] == player && grid[1][1] == player && grid[2][2] == "none") {
        winLocation = [2, 2]
        return true;
    } else if (grid[0][2] == player && grid[1][1] == player && grid[2][0] == "none") {
        winLocation = [2, 0]
        return true;
```

```
192         }
193         return false;
194     }
195     function easyGame() {
196         if (gameType == 1 && currentPlayer == 'o' && allowed) {
197             document.getElementById('message').innerHTML = "Thinking...";
198             setTimeout( //timeout to create illusion of thinking
199                 function(){
200                     if (nextWin('o')) { //if AI will win
201                         grid[winLocation[0]][winLocation[1]] = 'o';
202                         document.getElementById(winLocation[0].toString() +
                            winLocation[1].toString()).src = "o.png";
203                     } else if (nextWin('x')) { //block player from winning
204                         grid[winLocation[0]][winLocation[1]] = 'o';
205                         document.getElementById(winLocation[0].toString() +
                            winLocation[1].toString()).src = "o.png";
206                     } else {
207                         //if blocking or winning unnecessary, pick a random spot
208                         var x, y;
209                         var worked = false;
210                         while (!worked) { //pick random spots until one works
211                             x = Math.floor(Math.random() * 3);
212                             y = Math.floor(Math.random() * 3);
213                             if (grid[x][y] == "none") {
214                                 grid[x][y] = 'o';
215                                 document.getElementById(x.toString() + y.toString()).src =
                                    "o.png";
216                                 worked = true;
217                             } else {
218                                 worked = false;
219                             }
220                         }
221                     }
222                     //switching turn
223                     currentPlayer = 'x';
224                     document.getElementById('message').innerHTML = "Your turn.";
225                     //check if AI won
226                     if(winner('o')) {
227                         document.getElementById('message').innerHTML = "I win!<br /><button
                            onclick='newGame()'>New Game</button>";
228                         oWins++;
229                         updateScoreboard()
230                         allowed = false;
231                     } else if (winner('o') == undefined) {
232                         document.getElementById('message').innerHTML = "Tie!<br /><button
                            onclick='newGame()'>New Game</button>";
233                         allowed = false;
234                     }
235                 }, 700
236             )
237         }
238     }
```

index.htm

```
<html>
<head>
    <title>Tic-Tac-Toe</title>
    <script src="game.js"></script>
    <style>
        h2 {font-family: calibri; font-size: 36; margin-bottom: 0px;}
        h3 {font-family: calibri; font-size: 26; margin-bottom: 0px;}
        p {font-family: calibri; font-size: 16;}

        /* button CSS from
        https://www.w3schools.com/css/tryit.asp?filename=trycss_buttons_group_vertical */
        .btn-group .button {
          background-color: #4CAF50; /* Green */
          border: 1px solid green;
          color: white;
          padding: 15px 32px;
          text-align: center;
          text-decoration: none;
          display: inline-block;
          font-size: 16px;
          cursor: pointer;
          width: 150px;
          display: block;
        }

        .btn-group .button:not(:last-child) {
          border-bottom: none; /* Prevent double borders */
        }

        .btn-group .button:hover {
          background-color: #3e8e41;
        }
    </style>
    <link rel="icon" href="icon.png" type="image/x-icon" />
</head>
<body onload="resetGame()">
    <table width=100% height=100% rules=cols>
        <tr>
            <td width=25% valign=center rowspan=2>
                <table align=center>
                    <tr>
                        <td colspan=2><h2><u>Wins</u></h2></td>
                    </tr>
                    <tr align=center>
                        <td><h3>X</h3></td>
                        <td><h3>O</h3></td>
                    </tr>
                    <tr align=center>
                        <td><p id="xWins">0</p></td>
                        <td><p id="oWins">0</p></td>
                    </tr>
                </table>

                <h2 align=center>Start Tic-Tac-Toe</h2>
                <!-- button group from
                https://www.w3schools.com/css/tryit.asp?filename=trycss_buttons_group_vertical -->
                <div class="btn-group" align=center>
                    <button onclick="startGame(1)" class=button>1 Player</button>
                    <button onclick="startGame(0)" class=button>2 Players</button>
                </div>
            </td>
            <td align=center>
                <table rules=all>
                    <tr>
                        <td>
```

```
                            <img id="00" src="none.png"
                            onclick="playTurn(0,0,currentPlayer)" />
                        </td>
                        <td>
                            <img id="01" src="none.png"
                            onclick="playTurn(0,1,currentPlayer)">
                        </td>
                        <td>
                            <img id="02" src="none.png"
                            onclick="playTurn(0,2,currentPlayer)">
                        </td>
                    </tr>
                    <tr>
                        <td>
                            <img id="10" src="none.png"
                            onclick="playTurn(1,0,currentPlayer)">
                        </td>
                        <td>
                            <img id="11" src="none.png"
                            onclick="playTurn(1,1,currentPlayer)">
                        </td>
                        <td>
                            <img id="12" src="none.png"
                            onclick="playTurn(1,2,currentPlayer)">
                        </td>
                    </tr>
                    <tr>
                        <td>
                            <img id="20" src="none.png"
                            onclick="playTurn(2,0,currentPlayer)">
                        </td>
                        <td>
                            <img id="21" src="none.png"
                            onclick="playTurn(2,1,currentPlayer)">
                        </td>
                        <td>
                            <img id="22" src="none.png"
                            onclick="playTurn(2,2,currentPlayer)">
                        </td>
                    </tr>
                </table>
            </td>
        </tr>
        <tr height=20%>
            <td align=center><h2 id="message"> </h2></td>
        </tr>
    </table>
</body>
</html>
```