

## CpSc 130 Project 2 Spring, 2016; Dr. Conlon

Create an HTML/CSS/JavaScript Web page that plays tic-tac-toe with the user.

Using HTML and CSS, create a tic-tac-toe board as a table, with no borders on the outer edges. Next to the board should be a *Play Again* button, which, when clicked, will clear the board and re-initialize all settings for a new game.

Use graphics for the X's and O's. Clicking on a cell should cause an X graphic to be displayed in the cell, unless there is already an X or O there. Each time it is the computer's turn and there are two identical letters in a row, column, or diagonal with an empty third cell, the computer should play the empty cell. Otherwise, the computer should play an empty cell chosen randomly. As soon as either player has three in a row, the program should stop the game and announce the winner on the page, not with an *alert()*, and in a large colorful font.

Be sure to include header comments and follow indentation rules. Put as much of your JavaScript as possible into the `<head>` section. Indentation rules will be enforced strictly in JavaScript and CSS sections. The rubric for this project is on the reverse of this sheet. Do not lose this sheet, since you must submit it with your program listing.

This project is due by 3 p.m. on Wednesday, April 13, 2016. Submit your code listing with the rubric stapled so that the staple goes through the upper left of the listing and the upper right of the rubric, rubric side up. Place **all** files for the Web page in your `projects/project2` folder on the "P" drive. Your HTML file should be named with your last name and first initial, lower case, no spaces, with the extension `.html`. Thus, Maria Smith's project file would be called `smithm.html`.



### Suggested Order for Attacking the Problem

1. Create a basic 3x3 table.
2. Put an "onclick" in each cell, so that clicking will insert an X or O.
3. Add code to ignore clicks on cells that already have an X or O. *Level one.*
4. Add code to do the computer's move (O) at random after each user X.
5. Add code to detect and announce a win.
6. Add the *Play Again* button; give it an onclick event handler that clears all the cells, and clears variables to the game-start state.
7. Pretty-up the page formatting. *Level two.*
8. Make the computer smarter, looking for rows, columns, or diagonals with two O's and an empty space for a winning move, or with two X's for a blocking move, before choosing a random move. *Level three.*