CS 110: Homework 5 (5 Points)
Tic Tac Toe

LEARNING OBJECTIVES
- Practice working with loops
- Practice working with arrays and/or dictionaries
- Practice with logic
- Practice writing more complex programs

FILES TO DOWNLOAD
- https://eecs110.github.io/spring2019/course-files/homework/hw05.zip
- Demo Video

In this assignment, you will be making a game of tic tac toe, where a player will be playing the computer. Please write a program that accomplishes the following tasks:

1. Ask the user whether they want to be **x** or **o** (and ensure your program honors their choice).
2. Ask the user if they want to go first (and ensure your program honors their choice)
3. Prompt the user for their move (they'll enter a number from 1-9), which corresponds to where they want to go (as displayed below):

$$7 \mid 8 \mid 9$$
$$\overline{}$$
$$4 \mid 5 \mid 6$$
$$\overline{}$$
$$1 \mid 2 \mid 3$$

   a. If the user does not enter a number from 1-9, tell them their entry was invalid and that they should try again (use try/except as well as if/else).
   b. If the user enters a valid number that is not empty, tell them that slot is already taken.
4. If it's the computer's turn, have the computer randomly select a square, given the available choices (using the random module).
5. After each move, be sure to redraw the game board
6. If there is a win / loss / tie, print an appropriate message and terminate the game

Caveats

There are many solutions on the Internet for this assignment. DO NOT USE THEM. We know what they are, and can tell if you copied all or parts of them. We want you to invent your own strategy, given what you have learned. You get a 'do over,' so use your own ingenuity. Like all things, you will get out of this assignment what you put into it.

Extra Credit Options (up to 2 points total)

We highly recommend that you pursue these optional enhancements, to encourage you to think about one way (of many) that 'artificial intelligence' might work:

1. Make your computer 'smart' so that the computer always selects the optimal move. To do this, consider how the computer could:
    a. (1 point) Always block the user from winning (when possible)
    b. (1 point) If the computer is about to win, make sure it selects the correct square!
    c. (1 points): Other enhancements (see strategy discussion):
       https://www.wikihow.com/Win-at-Tic-Tac-Toe

2. (1 point) Draw a line through the winning pattern on the game board.

Checklist (Before you Submit)

Before you submit, make sure you've tested that your program does the following:

1	Allows the user to select whether they want to be x or o
2	Allows the user to indicate whether or not they want to go first
3	Program honors the user's selection (1-9) and places the symbol correctly
4	If the user enters something other than 1-9, the program tells the user what they did wrong and re-prompts them Note: consider using try/catch + continue (as we did in the beginning of Lecture 13)
5	Computer move does not overwrite the user's move
6	User's move does not overwrite computer's move
7	If the user and / or computer wins, the program ends and tells the user which player won
8	If there's a tie, the program ends and displays a "tie" message