Cisco, python essentials 1

Creating a tic-tac-toe game:

PROJECT Tic-Tac-Toe

Scenario

Your task is to write **a simple program that pretends to play *tic-tac-toe* with the user**. To make it all easier for you, we've decided to simplify the game. Here are our assumptions:

- the computer (i.e., your program) should play the game using 'X's;
- the user (e.g., you) should play the game using 'O's;
- the first move belongs to the computer − it always puts its first 'X' in the middle of the board;
- all the squares are numbered row by row starting with 1 (see the example session below for reference)
- the user inputs their move by entering the number of the square they choose − the number must be valid, i.e., it must be an integer, it must be greater than 0 and less than 10, and it cannot point to a field that is already occupied;

- the program checks if the game is over − there are four possible verdicts: the game should continue, the game ends with a tie, you win, or the computer wins;
- the computer responds with its move and the check is repeated;
- don't implement any form of artificial intelligence − a random field choice made by the computer is good enough for the game.

The example session with the program may look as follows:

```
+-------+-------+-------+
|       |       |       |
|   1   |   2   |   3   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   4   |   X   |   6   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   7   |   8   |   9   |
|       |       |       |
+-------+-------+-------+
Enter your move: 1
+-------+-------+-------+
|       |       |       |
|   O   |   2   |   3   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   4   |   X   |   6   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   7   |   8   |   9   |
|       |       |       |
+-------+-------+-------+
+-------+-------+-------+
|       |       |       |
|   O   |   X   |   3   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   4   |   X   |   6   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   7   |   8   |   9   |
|       |       |       |
+-------+-------+-------+
Enter your move: 8
+-------+-------+-------+
|       |       |       |
|   O   |   X   |   3   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   4   |   X   |   6   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   7   |   O   |   9   |
|       |       |       |
+-------+-------+-------+
+-------+-------+-------+
|       |       |       |
|   O   |   X   |   3   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   4   |   X   |   X   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   7   |   O   |   9   |
|       |       |       |
+-------+-------+-------+
Enter your move: 4
+-------+-------+-------+
|       |       |       |
|   O   |   X   |   3   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   O   |   X   |   X   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   7   |   O   |   9   |
|       |       |       |
+-------+-------+-------+
+-------+-------+-------+
|       |       |       |
|   O   |   X   |   X   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   O   |   X   |   X   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   7   |   O   |   9   |
|       |       |       |
+-------+-------+-------+
Enter your move: 7
```

```
+-------+-------+-------+
|       |       |       |
|   O   |   X   |   X   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   O   |   X   |   X   |
|       |       |       |
+-------+-------+-------+
|       |       |       |
|   O   |   O   |   9   |
|       |       |       |
+-------+-------+-------+
You won!
```

Requirements

Implement the following features:

- the board should be stored as a three-element list, while each element is another three-element list (the inner lists represent rows) so that all of the squares may be accessed using the following syntax:

```
board[row][column]
```

- each of the inner list's elements can contain 'O', 'X', or a digit representing the square's number (such a square is considered free)
- the board's appearance should be exactly the same as the one presented in the example.
- implement the functions defined for you in the editor.

Drawing a random integer number can be done by utilizing a Python function called `randrange().` The example program below shows how to use it (the program prints ten random numbers from 0 to 8).

Note: the from-import instruction provides access to the `randrange` function defined within an external Python module called random.

Course source: https://skillsforall.com/.

Remade by: **xbits.**