TIC TAC TOE CODE

```
import random

# creating a playing board
rows, cols = (3, 3)
play_board = [[0 for i in range(1, 4)] for j in range(1, 4)]
a = 1
for i in range(len(play_board)):
    for j in range(len(play_board)):
        play_board[i][j] = str(a)
        a += 1
# computer gets first move by random choice
computer_x = random.choice([0, 1, 2])
computer_y = random.choice([0, 1, 2])
# print(computer_x)
# print(computer_y)
play_board[computer_x][computer_y] = "X"
# we need count as the board has 9 fields - setting it to 1 as computer has first move
count = 1

def DisplayBoard(board):
    # the function accepts one parameter containing the board's current status
    # and prints the board out to the console
    for i in range(len(board)):
        print("-" * 25)
        print("|       |       |       |")
        for j in range(len(board)):
            if j == len(board) - 1:
                print(f"|   {board[i][j]}   ", end="|")
            else:
                print(f"|   {board[i][j]}   ", end="")
        print()
        print("|       |       |       |")
    print("-" * 25)


def EnterMove(board):
    # print(board)
    # print(board[0])
    # print(board[1])
    # print(board[2])
    # the function accepts the board current status, then asks the user about their move,
    # checks the user input and updates the board field, users moves are marked with "O"
    player_move = str(input("Select your move. Check the board and type a field number for your move:
"))
    for fields in board:
        for field in fields:
```

```
            # print(field, player_move)
            if field == player_move:
                i = board.index(fields)
                j = fields.index(field)
                board[i][j] = "O"
                return
    print("Field taken.")
    return EnterMove(board)


def VictoryFor(board, sign):
    global count
    print(sign)
    # the function checks the board status in order to check who won the game, computer using 'X' or the
player using 'O'

    # print(count)
    # for fields in board:
    #     for field in fields:
    #         if not field.isnumeric():
    #             count += 1
    count += 1
    # print("Printing count: " + str(count))
    for field in board:
        if field.count(sign) == 3:
            if sign == "O":
                print("You win!!!")
            else:
                print("You lose!")
            return False
        if board[0][0] == board[1][0] and board[0][0] == board[2][0]:
            if board[0][0] == "O":
                print("You win!!!")
            else:
                print("You lose!")
            return False
        if board[0][1] == board[1][1] and board[0][1] == board[2][1]:
            if board[0][1] == "O":
                print("You win!!!")
            else:
                print("You lose!")
            return False
        if board[0][2] == board[1][2] and board[0][2] == board[2][2]:
            if board[0][2] == "O":
                print("You win!!!")
            else:
                print("You lose!")
            return False
```

```
        if board[0][0] == board[1][1] and board[0][0] == board[2][2]:
            if board[0][0] == "O":
                print("You win!!!")
            else:
                print("You lose!")
            return False
        if board[0][2] == board[1][1] and board[0][2] == board[2][0]:
            if board[0][2] == "O":
                print("You win!!!")
            else:
                print("You lose!")
            return False
        if board[0][0] == board[0][1] and board[0][0] == board[0][2]:
            if board[0][0] == "O":
                print("You win!!!")
            else:
                print("You lose!")
            return False
        if board[1][0] == board[1][1] and board[1][0] == board[1][2]:
            if board[1][0] == "O":
                print("You win!!!")
            else:
                print("You lose!")
            return False
        if board[2][0] == board[2][1] and board[2][0] == board[2][2]:
            if board[2][0] == "O":
                print("You win!!!")
            else:
                print("You lose!")
            return False

    # if count is 9 and there was no winner up to this point then it is a draw
    if count == 9:
        print("No winners, it's a draw!!!")
        return False
    return True


def DrawMove(board):
    # the function draws the computer's random move and updates the board
    random_numbers = []
    for fields in board:
        for field in fields:
            if field.isnumeric():
                random_numbers.append(field)
    if len(random_numbers) > 1:
        computer_move = str(random.choice(random_numbers))
    else:
```

```
        computer_move = str(random_numbers[0])
    for fields in board:
        for field in fields:
            if field == computer_move:
                i = board.index(fields)
                j = fields.index(field)
                board[i][j] = "X"
    print("Computer plays on field number: " + computer_move)
    return True


# here we are displaying the board first time with the computer first move already in the middle of the
board
DisplayBoard(play_board)

game_is_on = True
# marks start the game, user turn to enter the move
while game_is_on:
    EnterMove(play_board)
    DisplayBoard(play_board)
    game_is_on = VictoryFor(play_board, "O")
    if not game_is_on:
        break
    game_is_on = DrawMove(play_board)
    if not game_is_on:
        break
    DisplayBoard(play_board)
    game_is_on = VictoryFor(play_board, "X")
```