

Tic Tac Toe

~scurtă documentație~

Acesta este primul meu joc în Python și am folosit tot ce am învățat până acum. Mai exact la realizarea acestui joc am folosit funcții si metode asociate. Acesta poate fi jucat în consolă și necesită doi jucători.

Principii ale jocului

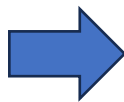
1. Doi jucători stau la același calculator și joacă.
2. Tabela trebuie afișată de fiecare data cand un jucător face o mișcare.
3. La fiecare mișcare trebuie să se accepte un input,apoi să se plaseze un simbol pe tabelă.

Fundamentul jocului

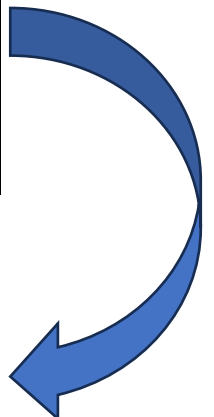
Jocul se bazează pe un “Numpad” care fiecare cifră de pe el reprezintă o posibilă mișcare.

7	8	9
4	5	6
1	2	3

Astfel,de exemplu, dacă jucătorul 1 alege să pună ‘X’ pe poziția 4 atunci tabela se va schimba,iar acolo se va afla ‘X’ și de asemenea dacă jucătorul 2 alege să pună ‘O’ pe poziția 5 se urmărește aceeași idee și așa mai departe până la finalul jocului.



X		



X	O	

Etapele principale ale realizării jocului

Etapa 1)

Scrierea unei funcții care afișează tabela de joc, corespunzătoare unui 'Numpad', cu indecși de la 1-9. Tabela are dimensiunea de 3x3.

```
6 usages
def display_board(board):
    print('\n'*100)
    print('  |  | ')
    print('  ' + board[7] + ' | ' + board[8] + ' | ' + board[9])
    print('  |  | ')
    print('-----')
    print('  |  | ')
    print('  ' + board[4] + ' | ' + board[5] + ' | ' + board[6])
    print('  |  | ')
    print('-----')
    print('  |  | ')
    print('  ' + board[1] + ' | ' + board[2] + ' | ' + board[3])
    print('  |  | ')
1 usage
```

Etapa 2)

Această funcție ajută la acceptarea de input-uri de la utilizator și permite jucătorului 1 să aleagă ce dorește să joace.

```
1 usage
def player_input():
    mark='ceva'
    while mark!='X' and mark!='O':
        mark=input('Player 1: Do you want to be X or O? ').upper()
        if mark=='X':
            return('X','O')
        else:
            return('O','X')
```

Etapa 3)

Una dintre cele mai importante funcții, permite asignarea poziției de pe tabela de joc ,a mișcării curente date de un jucător.

```
2 usages
def place_marker(board, mark, position):
    board[position]=mark
```

Etapa 4)

Toate posibilitățile de a câștiga a unui jucător(pe linie,pe coloană sau pe una dintre cele două diagonale).

```
2 usages
def win_check(board, mark):
    return ((board[7] == mark and board[8] == mark and board[9] == mark) or
            (board[4] == mark and board[5] == mark and board[6] == mark) or
            (board[1] == mark and board[2] == mark and board[3] == mark) or
            (board[7] == mark and board[4] == mark and board[1] == mark) or
            (board[8] == mark and board[5] == mark and board[2] == mark) or
            (board[9] == mark and board[6] == mark and board[3] == mark) or
            (board[7] == mark and board[5] == mark and board[3] == mark) or
            (board[9] == mark and board[5] == mark and board[1] == mark))
```

Etapa 5)

Folosind biblioteca random ,această funcție alege ,într-un mod aleator, ce jucător începe jocul.

```
1 usage
def choose_first():
    if random.randint(a: 0, b: 1) == 0:
        return 'Player 2'
    else:
        return 'Player 1'
```

Etapa 6+7)

Folosind o funcție ajutătoare, se determină dacă tabela de joc este plină sau nu.

```
2 usages
def space_check(board, position):
    return board[position] == ' '

2 usages
def full_board_check(board):
    for i in range(0,10):
        if space_check(board,i):
            return False
        else:
            return True
```

Etapa 8)

Funcție care indică alegerea mișcării jucătorului și verifică dacă mișcarea se poate realiza.

```
def player_choice(board):
    position = 0

    while position not in [1, 2, 3, 4, 5, 6, 7, 8, 9] or not space_check(board, position):
        position = int(input('Choose your next position: (1-9) '))

    return position
```

Etapa 9)

Dacă se dorește o nouă partidă de joc.

```
1 usage
def replay():
    return input('Do you want to play again? Enter Yes or No: ').lower().startswith('y')
```

Etapa 10)

Jocul propriu zis(funcția finală).

```
print('Welcome to Tic Tac Toe!')

while True:
    theBoard = [' '] * 10
    player1_mark, player2_mark = player_input()
    turn = choose_first() #se decide jucătorul care începe
    print(turn + ' will go first.')
    play_game = input('Are you ready to play? Enter Yes or No.')

    if play_game.lower()[0] == 'y': #dacă jucătorii sunt pregătiți atunci
                                    #se începe jocul
        game_on = True
    else:
        game_on = False

    while game_on:
        #în funcție de rândul jucătorului se face mișcarea
        if turn == 'Player 1':
            display_board(theBoard)
            position = player_choice(theBoard)
            place_marker(theBoard, player1_mark, position) #realizarea
#mișcării propriu zise a jucătorului 1

            #verificarea condițiilor de câștig
            if win_check(theBoard, player1_mark):
                display_board(theBoard)
                #dacă s-a îndeplinit unde dintre ele atunci se afisează un
#mesaj pe ecran și se oprește jocul
                print('Congratulations! You have won the game!')
                game_on = False
            else:
                #în cazul în care tabela este plină,dar nu s-a îndeplinit
                #nici o condiție de câștig atunci este remiză
                if full_board_check(theBoard):
                    display_board(theBoard)
                    print('The game is a draw!')
                    break
                else:
                    turn = 'Player 2'
        else:
            #analog si pentru jucătorul 2
            display_board(theBoard)
            position = player_choice(theBoard)
            place_marker(theBoard, player2_mark, position)

            if win_check(theBoard, player2_mark):
                display_board(theBoard)
                print('Player 2 has won!')
                game_on = False
            else:
                if full_board_check(theBoard):
                    display_board(theBoard)
                    print('The game is a draw!')
                    break
                else:
                    turn = 'Player 1'
```

```
if not replay():
    #în caz de că vrea să se reia jocul
    break
```

Câteva imagini sugestive

```
Welcome to Tic Tac Toe!
Player 1: Do you want to be X or O? X
Player 2 will go first.
Are you ready to play? Enter Yes or No.Yes
```

S-a selectat poziția 3, iar poziția 4 v-a fi următoarea.

-----	-----
	X
-----	-----
	O
Choose your next position: (1-9) 4	Choose your next position: (1-9)

După cateva mișcări jucătorul 2 a câștigat și se poate observa mesajul de rejucare.

X

X

O O O
Player 2 has won!
Do you want to play again? Enter Yes or No:

Do you want to play again? Enter Yes or No: Yes
Player 1: Do you want to be X or O?

În caz afirmativ, se reia tot jocul de la început.