

Tic tac Toe

```
board = [' ']*9
def display_board(board):
    print('      |      |')
    print('    '+board[0]+' | '+board[1]+' | '+board[2]+' ')
    print('      |      |')
    print('_____')
    print('      |      |')
    print('    '+board[3]+' | '+board[4]+' | '+board[5]+' ')
    print('      |      |')
    print('_____')
    print('      |      |')
    print('    '+board[6]+' | '+board[7]+' | '+board[8]+' ')
    print('      |      | \n')

def check_win(player_mark, board):
    ## If the player has won then there must be 3 consecutive Player values
    return (
        (board[0]==board[1]==board[2]==player_mark) or
        (board[3]==board[4]==board[5]==player_mark) or
        (board[6]==board[7]==board[8]==player_mark) or
        (board[0]==board[3]==board[6]==player_mark) or
        (board[1]==board[4]==board[7]==player_mark) or
        (board[2]==board[5]==board[8]==player_mark) or
        (board[0]==board[4]==board[8]==player_mark) or
        (board[2]==board[4]==board[6]==player_mark)
    )

def check_draw(board):
    return ' ' not in board

def board_copy(board):
    dupeBoard = []
    for j in board:
        dupeBoard.append(j)
    return dupeBoard

def test_win_move(board, player_mark, move):
    bCopy = board_copy(board)
    bCopy[move] = player_mark
    return check_win(player_mark, bCopy)

def win_strategy(board):
    for i in [0, 2, 6, 8]:
        if board[i] == ' ':
            return i
    # Play center
    if board[4] == ' ':
        return 4
    # Play a side
    for i in [1, 3, 5, 7]:
        if board[i] == ' ':
            return i
```

```

def fork_move(board, player_marker, move):
    # Determines if a move opens up a fork
    bCopy = board_copy(board)
    bCopy[move] = player_marker
    winning_moves = 0
    for j in range(0, 9):
        if test_win_move(bCopy, player_marker, j) and bCopy[j] == ' ':
            winning_moves += 1
    return winning_moves >= 2

def get_agent_move(board):
    # Return agent move with your strategy
    for i in range(0, 9):
        if board[i] == ' ' and test_win_move(board, 'X', i):
            return i
    # Check player win moves
    for i in range(0, 9):
        if board[i] == ' ' and test_win_move(board, 'O', i):
            return i

    for i in range(0, 9):
        if board[i] == ' ' and fork_move(board, 'X', i):
            return i

    for i in range(0, 9):
        if board[i] == ' ' and fork_move(board, 'O', i):
            return i
    return win_strategy(board)

def tictactoe():
    ### Note you need to recreate your board again here if you wish to play
    the game more than once
    Playing = True
    while Playing:
        InGame = True
        board = [' '] * 9
        print('Would you like to go first or second? (1/2)')
        if input() == '1':
            playerMarker = 'O'
        else:
            playerMarker = 'X'
            display_board(board)

        while InGame:
            if playerMarker == 'O':
                print('Player go: (0-8)')
                move = int(input())
                if board[move] != ' ':
                    print('Invalid move!')
                    #continue
            else:
                move = get_agent_move(board)
                board[move] = playerMarker
            if check_win(playerMarker, board):
                InGame = False

```

```

        display_board(board)
        if playerMarker == '0':
            print('Player wins!')
        else:
            print('Agent wins!')
        continue
    if check_draw(board):
        InGame = False
        display_board(board)
        print('It was a draw!')
        continue
    display_board(board)
    if playerMarker == '0':
        playerMarker = 'X'
    else:
        playerMarker = '0'

print('Type y to keep playing')
inp = input()
if inp != 'y' and inp != 'Y':
    Playing = False

```

OUTPUT

Would you like to go first or second? (1/2)

1

Player go: (0-8)

5

```

|---|---|---|
|   |   |   |
|---|---|---|
|   |   |0  |
|---|---|---|
|   |   |   |
|---|---|---|

```

```

|---|---|---|
|X  |   |   |
|---|---|---|
|   |   |0  |
|---|---|---|
|   |   |   |
|---|---|---|

```

Player go: (0-8)

8

```

|---|---|---|
|X  |   |   |
|---|---|---|
|   |   |0  |
|---|---|---|
|   |   |0  |
|---|---|---|

```

---	---	---
X		X
---	---	---
		0
---	---	---
		0
---	---	---

Player go: (0-8)

1

---	---	---
X	0	X
---	---	---
		0
---	---	---
		0
---	---	---

---	---	---
X	0	X
---	---	---
		0
---	---	---
X		0
---	---	---

Player go: (0-8)

4

---	---	---
X	0	X
---	---	---
	0	0
---	---	---
X		0
---	---	---

---	---	---
X	0	X
---	---	---
X	0	0
---	---	---
X		0
---	---	---

Agent wins!

Type y to keep playing