import numpy as np

def create\_board():

    return np.array([[" ", " ", " "],

                     [" ", " ", " "],

                     [" ", " ", " "]])

def coordinates(board, player):

    i, j, cn = (-1, -1, 0)

    player\_symbol = 'X' if player == 1 else 'O'

    while (i < 0 or i > 2 or j < 0 or j > 2) or (board[i][j] != " "):

        if cn > 0:

            print("Wrong Input! Try Again")

        print("Player {}'s turn ({})".format(player, player\_symbol))

        i = int(input("x-coordinates (1-3): ")) - 1

        j = int(input("y-coordinates (1-3): ")) - 1

        cn += 1

    board[i][j] = player\_symbol

    return board

def row\_win(board, player):

    player\_symbol = 'X' if player == 1 else 'O'

    for x in range(len(board)):

        if all(board[x, y] == player\_symbol for y in range(len(board))):

            return True

    return False

def col\_win(board, player):

    player\_symbol = 'X' if player == 1 else 'O'

    for x in range(len(board)):

        if all(board[y][x] == player\_symbol for y in range(len(board))):

            return True

    return False

def diag\_win(board, player):

    player\_symbol = 'X' if player == 1 else 'O'

    if all(board[x][x] == player\_symbol for x in range(len(board))):

        return True

    if all(board[x][len(board)-1-x] == player\_symbol for x in range(len(board))):

        return True

    return False

def evaluate(board):

    winner = 0

    for player in [1, 2]:

        if (row\_win(board, player) or

                col\_win(board, player) or

                diag\_win(board, player)):

            winner = player

    if np.all(board != " ") and winner == 0:

        winner = -1

    return winner

def play\_game():

    board, winner, counter = create\_board(), 0, 1

    print(board)

    while winner == 0:

        for player in [1, 2]:

            board = coordinates(board, player)

            print("Board after " + str(counter) + " move:")

            print(board)

            counter += 1

            winner = evaluate(board)

            if winner != 0:

                break

    return winner

winner = play\_game()

if winner == -1:

    print("It's a draw!")

else:

    print("Winner is Player {} ({})".format(winner, 'X' if winner == 1 else 'O'))







