LAB 1

Tic Tac Toe

Code:-

import random

def initialize\_board():

return [[' ' for \_ in range(3)] for \_ in range(3)]

def display\_board(board):

for row in board:

print('|'.join(row))

print('-' \* 5)

def check\_winner(board):

for row in board:

if row[0] == row[1] == row[2] != ' ':

return row[0]

for col in range(3):

if board[0][col] == board[1][col] == board[2][col] != ' ':

return board[0][col]

if board[0][0] == board[1][1] == board[2][2] != ' ':

return board[0][0]

if board[0][2] == board[1][1] == board[2][0] != ' ':

return board[0][2]

return None

def available\_moves(board):

return [(i, j) for i in range(3) for j in range(3) if board[i][j] == ' ']

def check\_two\_in\_a\_row(board, player):

for row in range(3):

if board[row].count(player) == 2 and board[row].count(' ') == 1:

return row, board[row].index(' ')

for col in range(3):

if [board[row][col] for row in range(3)].count(player) == 2:

empty\_index = [row for row in range(3) if board[row][col] == ' ']

if empty\_index:

return empty\_index[0], col

if [board[i][i] for i in range(3)].count(player) == 2:

empty\_index = [i for i in range(3) if board[i][i] == ' ']

if empty\_index:

return empty\_index[0], empty\_index[0]

if [board[i][2 - i] for i in range(3)].count(player) == 2:

empty\_index = [i for i in range(3) if board[i][2 - i] == ' ']

if empty\_index:

return empty\_index[0], 2 - empty\_index[0]

return None

def make\_move(board, player, move):

board[move[0]][move[1]] = player

def computer\_move(board):

move = check\_two\_in\_a\_row(board, 'O')

if move:

make\_move(board, 'O', move)

return

move = check\_two\_in\_a\_row(board, 'X')

if move:

make\_move(board, 'O', move)

return

moves = available\_moves(board)

if moves:

move = random.choice(moves)

make\_move(board, 'O', move)

def user\_move(board):

while True:

try:

row = int(input("Enter row (0-2): "))

col = int(input("Enter column (0-2): "))

if board[row][col] == ' ':

make\_move(board, 'X', (row, col))

return

else:

print("That spot is already taken. Try again.")

except (ValueError, IndexError):

print("Invalid input. Please enter numbers between 0 and 2.")

def play\_game():

board = initialize\_board()

players = ['X', 'O']

current\_player = 0

for \_ in range(9):

display\_board(board)

if current\_player == 0:

user\_move(board)

else:

computer\_move(board)

winner = check\_winner(board)

if winner:

display\_board(board)

print(f"Player {winner} wins!")

return

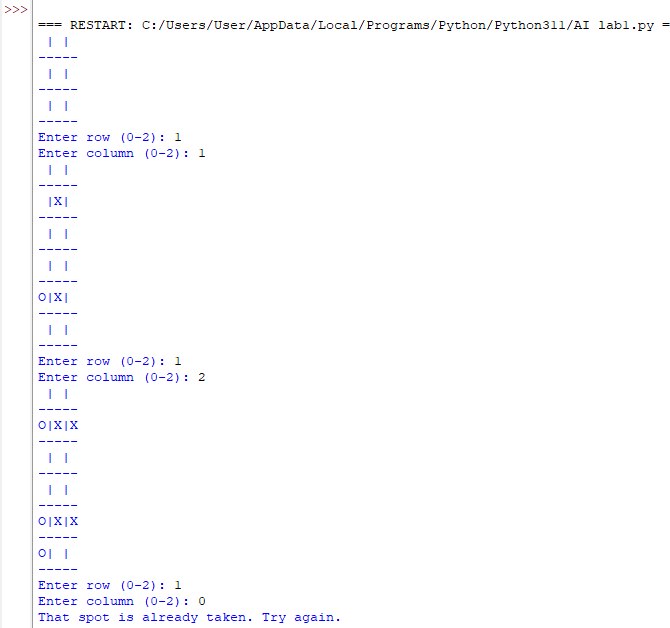
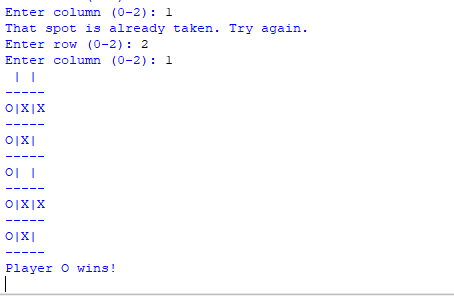
current\_player = 1 - current\_player

display\_board(board)

print("It's a draw!")

play\_game()

Output:-

  
  
Observation book:

