Aim:

Project Module.

Source Code:

CTP28132.py

```
def print_board(board):
    for row in board:
        print(" | ".join(row))
        print("-" * 5)

def check_winner(board, player):
    # Check rows, columns, and diagonals
    win_conditions = [
        [board[0][0], board[0][1], board[0][2]],
        [board[1][0], board[1][1], board[1][2]],
        [board[2][0], board[2][1], board[2][2]],
        [board[0][0], board[1][0], board[2][0]],
        [board[0][1], board[1][1], board[2][1]],
        [board[0][2], board[1][2], board[2][2]],
        [board[0][0], board[1][1], board[2][2]],
        [board[2][0], board[1][1], board[0][2]]
    ]
    return [player, player, player] in win_conditions

def is_draw(board):
    return all(cell != ' ' for row in board for cell in row)

def tic_tac_toe():
    board = [[' ' for _ in range(3)] for _ in range(3)]
    current_player = 'X'

    while True:
        print_board(board)
        print(f"Player {current_player}, make your move (row and column): ")
        row, col = map(int, input().split())

        if board[row][col] == ' ':
            board[row][col] = current_player
        else:
            print("Cell is already occupied. Try again.")
            continue

        if check_winner(board, current_player):
            print_board(board)
            print(f"Player {current_player} wins!")
            break

        if is_draw(board):
            print_board(board)
            print("It's a draw!")
            break
```

```
        current_player = 'O' if current_player == 'X' else 'X'

tic_tac_toe()
```

Execution Results - All test cases have succeeded!

Test Case - 1
User Output
Hello World
Hello World