

Sanjana Shetty

5-E

Tic Tac Toe Game:

```
board = [['-' for _ in range(3)] for _ in range(3)]

def is_board_filled(board):
    for row in board:
        for element in row:
            if element == '-':
                return False
    return True

def check_win(board, player):
    # Check rows
    for row in board:
        if all(element == player for element in row):
            return True

    # Check columns
    for col in range(len(board[0])):
        if all(board[row][col] == player for row in range(len(board))):
            return True

    # Check diagonals
    if all(board[i][i] == player for i in range(len(board))):
        return True
    if all(board[i][len(board) - i - 1] == player for i in
range(len(board))):
        return True

    return False

def show_board(board):
    for row in board:
        print(" ".join(row))
    print() # Add a blank line after the board

import random
```

```

def start_game(board):
    current_player = random.choice(['X', 'O'])
    while True:
        print(f"Player {current_player}'s turn.")

        try:
            row = int(input("Enter row (0-2): "))
            col = int(input("Enter column (0-2): "))
        except ValueError:
            print("Invalid input. Please enter integers between 0 and 2.")
            continue

        if 0 <= row <= 2 and 0 <= col <= 2:
            if board[row][col] == '-':
                board[row][col] = current_player

                show_board(board)

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

                if is_board_filled(board):
                    print("It's a draw!")
                    break

                current_player = 'O' if current_player == 'X' else 'X'
            else:
                print("Invalid move. Cell already occupied. Try again.")
        else:
            print("Invalid input. Please enter numbers between 0 and 2.")

# Start the game
show_board(board)
start_game(board)

```

OUTPUT:

- Draw

```
- - -  
- - -  
- - -  
  
Player O's turn.  
Enter row (0-2): 0  
Enter column (0-2): 0  
O - -  
- - -  
- - -  
  
Player X's turn.  
Enter row (0-2): 1  
Enter column (0-2): 1  
O - -  
- X -  
- - -  
  
Player O's turn.  
Enter row (0-2): 2  
Enter column (0-2): 2  
O - -  
- X -  
- - O  
  
Player X's turn.  
Enter row (0-2): 2  
Enter column (0-2): 0  
O - -  
- X -  
X - O  
  
Player O's turn.  
Enter row (0-2): 1  
Enter column (0-2): 0  
O - -  
O X -  
X - O  
  
Player X's turn.  
Enter row (0-2): 0  
Enter column (0-2): 1  
O X -  
O X -  
X - O  
  
Player O's turn.  
Enter row (0-2): 0  
Enter column (0-2): 2  
O X O  
O X -  
X - O  
  
Player X's turn.  
Enter row (0-2): 1  
Enter column (0-2): 2  
O X O  
O X X
```

2. Win

```
- - -  
- - -  
- - -  
  
Player X's turn.  
Enter row (0-2): 0  
Enter column (0-2): 0  
X - -  
- - -  
- - -  
  
Player O's turn.  
Enter row (0-2): 1  
Enter column (0-2): 0  
X - -  
O - -  
- - -  
  
Player X's turn.  
Enter row (0-2): 1  
Enter column (0-2): 1  
X - -  
O X -  
- - -  
  
Player O's turn.  
Enter row (0-2): 2  
Enter column (0-2): 1  
X - -  
O X -  
- O -  
  
Player X's turn.  
Enter row (0-2): 2  
Enter column (0-2): 2  
X - -  
O X -  
- O X  
  
Player X wins!
```

3. Occupying the same position

```
- - -  
- - -  
- - -  
  
Player X's turn.  
Enter row (0-2): 0  
Enter column (0-2): 0  
X - -  
- - -  
- - -  
  
Player O's turn.  
Enter row (0-2): 1  
Enter column (0-2): 2  
X - -  
- - O  
- - -  
  
Player X's turn.  
Enter row (0-2): 1  
Enter column (0-2): 1  
X - -  
- X O  
- - -  
  
Player O's turn.  
Enter row (0-2): 2  
Enter column (0-2): 2  
X - -  
- X O  
- - O  
  
Player X's turn.  
Enter row (0-2): 1  
Enter column (0-2): 1  
Invalid move. Cell already occupied. Try again.  
Player X's turn.  
Enter row (0-2): 
```