```
def print_board(board):
    for row in board:
        print(" | ".join(row))
        print("-" * 9)



def is_draw(board):
    for row in board:
        if ' ' in row:
            return False
    return True

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

    # Check columns
    for col in range(3):
        if all(board[row][col] == player for row in range(3)):
            return True

    # Check diagonals
    if all(board[i][i] == player for i in range(3)) or all(board[i][2 - i] == player for i in range(3)):
        return True

    return False

def best_move(board):
    # Placeholder for AI move
    for i in range(3):
        for j in range(3):
            if board[i][j] == ' ':
                return i, j

    return 0, 0  # Default move if the board is full

# The rest of your main function and code remains the same

if   name   == "  main  ":
```

```
main()
```

```
  |   |
---------
  |   |
---------
  |   |
---------
Enter row and column numbers from 0-2. Enter your move (row and column): 0 0
X | O |
---------
  |   |
---------
  |   |
---------
Enter row and column numbers from 0-2. Enter your move (row and column): 0 2
X | O | X
---------
O |   |
---------
  |   |
---------
Enter row and column numbers from 0-2. Enter your move (row and column): 1 1
X | O | X
---------
O | X | O
---------
  |   |
---------
Enter row and column numbers from 0-2. Enter your move (row and column): 2 1
X | O | X
---------
O | X | O
---------
O | X |
---------
Enter row and column numbers from 0-2. Enter your move (row and column): 0 1
Invalid move. Try again.
Enter row and column numbers from 0-2. Enter your move (row and column): 2 2
X | O | X
---------
O | X | O
---------
O | X | X
```

```
---------
You win!
```