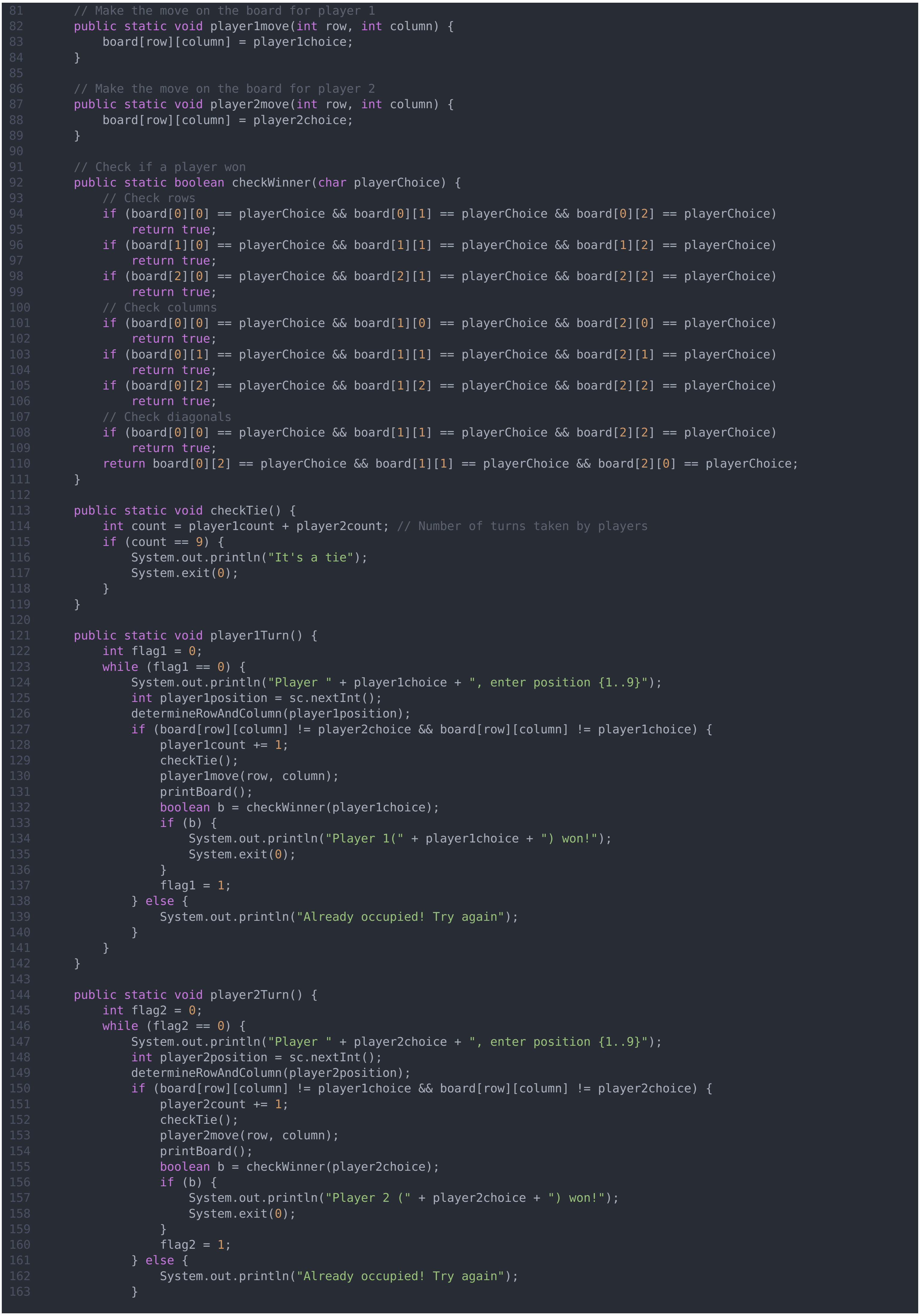
ticTacToe.java

```java
// TicTacToe game in Java for two players.

import java.util.Scanner;

public class ticTacToe {
    public static Scanner sc = new Scanner(System.in);
    static char[][] board = {
            {'1', '2', '3'},
            {'4', '5', '6'},
            {'7', '8', '9'},
    };
    static char player1choice, player2choice;
    static int row, column;
    static int player1count = 0, player2count = 0;

    public static void main(String[] args) {
        System.out.println("Welcome to the tictactoe game made by Kavin!");
        System.out.println("Enter letter to represent player 1: ");
        player1choice = sc.next().charAt(0);
        System.out.println("Enter letter to represent player 2: ");
        player2choice = sc.next().charAt(0);
        printBoard();
        // If no player has won, continue
        while (!checkWinner(player1choice) && !checkWinner(player2choice)) {
            player1Turn();
            player2Turn();
        }
    }

    // Print the current board
    public static void printBoard() {
        for (int i = 0; i < 3; i++) {
            System.out.print("| ");
            for (int j = 0; j < 3; j++) {
                System.out.print(board[i][j] + " | ");
            }
            System.out.println();
        }
    }

    // Determine 2D (x,y) coordinates from 1D position
    public static void determineRowAndColumn(int position) {
        if (position == 1) {
            row = 0;
            column = 0;
        }
        if (position == 2) {
            row = 0;
            column = 1;
        }
        if (position == 3) {
            row = 0;
            column = 2;
        }
        if (position == 4) {
            row = 1;
            column = 0;
        }
        if (position == 5) {
            row = 1;
            column = 1;
        }
        if (position == 6) {
            row = 1;
            column = 2;
        }
        if (position == 7) {
            row = 2;
            column = 0;
        }
        if (position == 8) {
            row = 2;
            column = 1;
        }
        if (position == 9) {
            row = 2;
            column = 2;
        }
    }

```



```java
    // Make the move on the board for player 1
    public static void player1move(int row, int column) {
        board[row][column] = player1choice;
    }

    // Make the move on the board for player 2
    public static void player2move(int row, int column) {
        board[row][column] = player2choice;
    }

    // Check if a player won
    public static boolean checkWinner(char playerChoice) {
        // Check rows
        if (board[0][0] == playerChoice && board[0][1] == playerChoice && board[0][2] == playerChoice)
            return true;
        if (board[1][0] == playerChoice && board[1][1] == playerChoice && board[1][2] == playerChoice)
            return true;
        if (board[2][0] == playerChoice && board[2][1] == playerChoice && board[2][2] == playerChoice)
            return true;
        // Check columns
        if (board[0][0] == playerChoice && board[1][0] == playerChoice && board[2][0] == playerChoice)
            return true;
        if (board[0][1] == playerChoice && board[1][1] == playerChoice && board[2][1] == playerChoice)
            return true;
        if (board[0][2] == playerChoice && board[1][2] == playerChoice && board[2][2] == playerChoice)
            return true;
        // Check diagonals
        if (board[0][0] == playerChoice && board[1][1] == playerChoice && board[2][2] == playerChoice)
            return true;
        return board[0][2] == playerChoice && board[1][1] == playerChoice && board[2][0] == playerChoice;
    }

    public static void checkTie() {
        int count = player1count + player2count; // Number of turns taken by players
        if (count == 9) {
            System.out.println("It's a tie");
            System.exit(0);
        }
    }

    public static void player1Turn() {
        int flag1 = 0;
        while (flag1 == 0) {
            System.out.println("Player " + player1choice + ", enter position {1..9}");
            int player1position = sc.nextInt();
            determineRowAndColumn(player1position);
            if (board[row][column] != player2choice && board[row][column] != player1choice) {
                player1count += 1;
                checkTie();
                player1move(row, column);
                printBoard();
                boolean b = checkWinner(player1choice);
                if (b) {
                    System.out.println("Player 1(" + player1choice + ") won!");
                    System.exit(0);
                }
                flag1 = 1;
            } else {
                System.out.println("Already occupied! Try again");
            }
        }
    }

    public static void player2Turn() {
        int flag2 = 0;
        while (flag2 == 0) {
            System.out.println("Player " + player2choice + ", enter position {1..9}");
            int player2position = sc.nextInt();
            determineRowAndColumn(player2position);
            if (board[row][column] != player1choice && board[row][column] != player2choice) {
                player2count += 1;
                checkTie();
                player2move(row, column);
                printBoard();
                boolean b = checkWinner(player2choice);
                if (b) {
                    System.out.println("Player 2 (" + player2choice + ") won!");
                    System.exit(0);
                }
                flag2 = 1;
            } else {
                System.out.println("Already occupied! Try again");
            }
        }
    }
}
/**
 * Variable             Data        Table
 * board                int[][]     Store board
 * player1choice        char        Store 1st users avatar
 * player2choice        char        Store 2nd users avatar
 * player1position      int         Where 1st user choose to move
 * player2position      int         Where 2nd user choose to move
 * row, column          int         2d coordinates for move
 * position             int         1d coordinates for move
 * i, j                 int         Used in calculation
 */
```